

**Diana Pérez Marín,  
J. Ángel Velázquez-Iturbide**

**Actas del VII Seminario de  
Investigación en Tecnologías de  
la Información Aplicadas a la  
Educación (SITIAE-2013)**

Número 2014-01

*Serie de Informes Técnicos DLSII-URJC*

**ISSN 1988-8074**

**Departamento de Lenguajes y Sistemas Informáticos I  
Universidad Rey Juan Carlos**

## Tabla de Contenidos

- 3 Prefacio**  
*Diana Pérez Marín, J. Ángel Velázquez-Iturbide*
- 4 Familias de Visualizaciones de Árboles de Recursión: Definición y Diseño**  
*J. Ángel Velázquez Iturbide, Antonio Pérez Carrasco*
- 18 El intercambio de papeles como estrategia docente**  
*Isidoro Hernán Losada*
- 33 Kinect con Intrael**  
*Francisco Domínguez Mateos*
- 43 GreedExCol: un sistema interactivo colaborativo para el aprendizaje de algoritmos voraces**  
*Maximiliano Paredes-Velasco, Ouafae Debdi, J. Ángel Velázquez-Iturbide*
- 59 Cuadernia: una buena alternativa para la creación de actividades educativas**  
*Liliana Patricia Santacruz Valencia*
- 72 Recommendations on the Design of Semi-Immersive Interactive Computer Educational Applications for Children**  
*Haydeé Artaza Alvarez, Diana Perez Marín*
- 91 Análisis de plagios en prácticas de programación**  
*Manuel Rubio Sánchez*

## Prefacio

El VII Seminario de Investigación en Tecnologías de la Información Aplicadas a la Educación se celebró durante los meses de abril y mayo del año 2013 con el objetivo de convertirse en un lugar de puesta en común de la investigación realizada por el grupo LITE y por las personas relacionadas con la Informática Educativa.

El contenido del seminario está orientado a estudiantes e investigadores interesados en la Informática Educativa. Al igual que su predecesor, SITIAE 2013 se vertebra en varias sesiones agrupadas por temáticas. Todas las sesiones se celebraron en el campus de Móstoles de la Universidad Rey Juan Carlos. Se invitó a los ponentes a publicar una versión extendida de su charla en esta Serie de Informes Técnicos del Departamento de Lenguajes y Sistemas Informáticos I de la Escuela Técnica Superior de Ingeniería Informática, recibiendo siete contribuciones de alta calidad que se han publicado en el número 2014-01 con el ISSN 1988-8074.

La primera contribución *“Familias de Visualizaciones de Árboles de Recursión: Definición y Diseño”* presenta la investigación realizada en análisis de visualizaciones propias de técnicas de diseño. La segunda contribución *“El intercambio de papeles como estrategia docente”* muestra la aplicación de una nueva metodología docente en la asignatura de Programación Orientada a Objetos. La tercera contribución *“Kinect con Intrael”* presenta las bases del uso y arquitectura de Kinect con Intrael. La cuarta contribución *“GreedExCol: un sistema interactivo colaborativo para el aprendizaje de algoritmos voraces”* revisa los principales estándares de aprendizaje, centrándose en el estándar IMS LD. La quinta contribución *“Cuadernia: una buena alternativa para la creación de actividades educativas”* propone la herramienta Cuadernia para la creación de actividades educativas atrayentes e interactivas. La sexta contribución *“Recommendations on the Design of Semi-Immersive Interactive Computer Educational Applications for Children”* proporciona un conjunto de recomendaciones para el diseño de sistemas semi-inmersivos educativos; y, la séptima contribución *“Análisis de plagios en prácticas de programación”* analiza la opinión de alumnos y profesores sobre el grado de severidad de diferentes escenarios relacionados con posibles copias/trampas/fraude académico y, presenta los resultados de aplicar herramientas de detección de copias en prácticas de Ingeniería Informática.

Muchas gracias a todos los autores por vuestras contribuciones, y a todos los ponentes por hacer de SITIAE 2013 todo un éxito con la participación de más de 50 estudiantes de Grados tan diversos como Educación Infantil, Primaria e Ingeniería Informática de la Universidad Rey Juan Carlos, y de profesores e investigadores del área. Se quiere agradecer también al Ministerio de Economía y Competitividad la financiación parcial de SITIAE 2013 con el proyecto TIN2011-29542-C02-01.

Diana Pérez Marín, J. Ángel Velázquez-Iturbide

## Familias de Visualizaciones de Árboles de Recursión: Definición y Diseño<sup>1</sup>

J. Ángel Velázquez Iturbide<sup>1</sup>, Antonio Pérez Carrasco<sup>1</sup>

<sup>1</sup> Departamento de Lenguajes y Sistemas Informáticos I, Universidad Rey Juan Carlos  
C/ Tulipán s/n, 28933 Mostoles, Madrid, España  
{angel.velazquez,antonio.perez.carrasco}@urjc.es

**Resumen.** Proponemos el concepto de familia de visualizaciones como un conjunto de visualizaciones que muestran el resultado de ejecutar sucesivas veces un algoritmo con valores consecutivos de los datos de entrada. Nuestro objetivo es mejorar la comprensión de la redundancia presente en numerosos algoritmos con recursividad múltiple. Las potenciales ventajas de una familia de visualizaciones son: mejor comprensión del algoritmo y más fácil análisis de su redundancia. También describimos cómo hacer un buen uso de las familias de visualizaciones, distinguiendo entre dos clases de algoritmos y aportando una guía informal para su diseño y construcción.

**Palabras clave:** Animación de programas, recursividad, árboles de recursión, casos de prueba.

### 1 Introducción

La visualización de programas [7] es un recurso muy utilizado para mejorar la comprensión de los programas, tarea difícil por su carácter abstracto. La animación de algoritmos [7] es similar pero con dos características distintivas: alto nivel de abstracción (mayor que el de código fuente) y carácter dinámico. Tanto la visualización

---

<sup>1</sup> Una versión preliminar este artículo se publicó como "Familias de visualizaciones de los árboles de recursión", J. Ángel Velázquez Iturbide y Antonio Pérez Carrasco, *SIIE13 XV International Symposium on Computers in Education – Proceedings*, Maria José Marcelino, Maria Cristina Azebedo Gomes y António José Mendes (eds.), 2013 (ISBN 978-989-96261-3-3), págs. 18-23.

de programas como la animación de algoritmos se han utilizado principalmente con fines docentes, habiéndose estudiado diversos problemas en su uso: eficacia educativa, adopción por los profesores, esfuerzo de creación, etc.

En este artículo nos basamos en una aproximación a la creación sin esfuerzo de animaciones de algoritmos mediante el uso de técnicas de visualización de programas, donde:

- La visualización de los estados relevantes de la ejecución de un algoritmo se genera como efecto colateral de la misma.
- El formato de las visualizaciones tiene el mismo nivel de abstracción que el diseño de un algoritmo.

Esta aproximación se ha implementado en el sistema SRec, que es un sistema diseñado para la visualización y animación de la recursividad [12]. SRec soporta varios formatos gráficos, destacando los árboles de recursión. También soporta una gran variedad de funciones de interacción [11].

En el artículo proponemos el concepto de familia de visualizaciones como un conjunto de visualizaciones que muestran el resultado de ejecutar sucesivas veces un algoritmo con valores consecutivos de los datos de entrada. Nuestro objetivo es mejorar la comprensión de la redundancia presente en numerosos algoritmos con recursividad múltiple. En la sección siguiente presentamos la propuesta, así como una guía para desarrollar las familias de visualizaciones en dos clases de algoritmos. La sección tercera relaciona las familias de visualizaciones con otros trabajos en animación de algoritmos. Finalmente presentamos nuestras conclusiones y un conjunto de cuestiones abiertas a trabajos futuros.

## 2 Familias de Visualizaciones de los Árboles de Recursión

Presentamos primero una motivación y la definición de familia de visualizaciones y después analizamos cómo concretar su creación para dos clases de algoritmos.

### 2.1 Motivación

Hay algoritmos muy ineficientes con recursividad múltiple. Esta situación se da cuando el número de subproblemas distintos es polinómico mientras que el número de subproblemas generados (es decir, de llamadas recursivas) es exponencial o factorial. La explicación de esta situación es que hay subproblemas que se generan (y resuelven) numerosas veces.

Encontramos esta situación en numerosos casos, destacando la formulación recursiva de los algoritmos de programación dinámica. Su diseño siempre incluye una primera fase de análisis de la redundancia a partir de sus árboles de recursión [1][6] y otra fase posterior para su eliminación. Como resultado, los algoritmos de

programación dinámica son eficientes en tiempo (es decir, tienen complejidad polinomial).

Veamos las dos dificultades principales que hay para el análisis de redundancia con árboles de recursión.

**Elección de los Datos de Entrada para el Análisis.** Sea la conocida serie de los números de Fibonacci en Java:

```
public static int fib (int n) {
    if (n==0 || n==1)
        return 1;
    else
        return fib(n-1) + fib(n-2);
}
```

La Figura 1 muestra su árbol de recursión para  $n=4$ . Cada nodo del árbol representa una llamada recursiva, con los parámetros en la mitad superior (en la figura, en azul) y el resultado en la inferior (en rojo). La raíz del árbol representa la llamada inicial. De cada nodo surgen tantos arcos a nodos del nivel inferior como llamadas recursivas realice. Los arcos aparecen de izquierda a derecha en orden cronológico de realización de sus llamadas recursivas correspondientes. En la figura, la llamada inicial tiene  $n=4$  como parámetro y finalmente dará un resultado de 5. Esta llamada inicial a su vez hace dos llamadas con parámetros sucesivos 3 y 2.

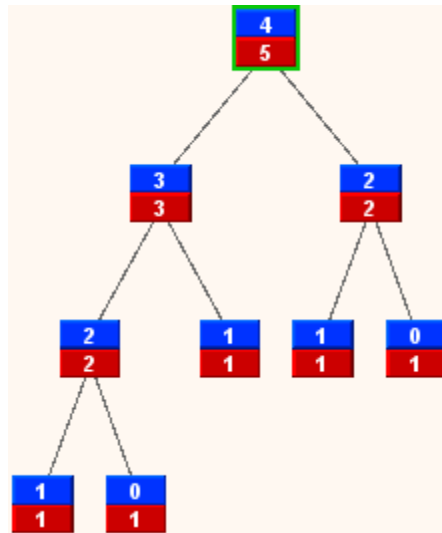


Fig. 1. Árbol de recursión de  $fib(4)$ .

En este árbol se observa que los casos básicos  $fib(0)$  y  $fib(1)$  son llamados 2 y 3 veces respectivamente, y el caso recursivo  $fib(2)$ , 2 veces. El árbol es suficiente para comprobar que hay redundancia puesto que hay un caso recursivo que se ejecuta más de una vez. Sin embargo, quizá convenga tener una evidencia mayor que la proporcionada por una sola llamada recursiva repetida. Por tanto, sería mejor dibujar el árbol de recursión para un parámetro mayor, como  $n=5$  o  $n=6$ .

**Comprensión del Algoritmo Recursivo.** El algoritmo anterior es de fácil comprensión pero no es éste el caso de los algoritmos recursivos planteados para problemas de optimización y que son la base para desarrollar algoritmos de programación dinámica. En este caso, los alumnos tienen dificultades para comprender las decisiones básicas de su diseño: organización de la construcción de la solución en etapas y las decisiones posibles a tomar en cada etapa.

Por ejemplo, sea el problema de la mochila 0/1 [2][4][8], que puede resolverse mediante el siguiente algoritmo recursivo:

```
public static int mochila01
    (int[] ps, int[] bs, int c) {
    return m(ps,bs,c,0,c);
}

private static int m
    (int[] ps, int[] bs, int c,
     int i, int p) {
    if (i==ps.length)
        return 0;
    else if (p<ps[i])
        return m(ps,bs,c,i+1,p);
    else
        return Math.max (m(ps,bs,c,i+1,p),
                        m(ps,bs,c,i+1,p-ps[i])+bs[i]);
}
```

donde *mochila01* es el método principal y *m* es el método recursivo auxiliar. Los parámetros que controlan el proceso recursivo son el índice *i* y el peso  $p_i$  de cada objeto.

El algoritmo recursivo va analizando los objetos de izquierda a derecha, habiendo por tanto tantas etapas como objetos. Si un objeto no cabe en la mochila, sólo hay una decisión posible: no meterlo. Sin embargo, si cabe, debe elegirse la decisión con beneficio máximo: meterlo o no meterlo en la mochila.

La Figura 2 muestra el árbol de recursión de este algoritmo para los datos  $ps=\{35,22,42,22,5,18\}$  y  $c=44$ . La figura permite ver las principales decisiones de diseño del algoritmo. La altura del árbol es proporcional al número de objetos. Asimismo, el número de descendientes de cada nodo hace explícito que en algunos casos puede considerarse una sola posibilidad y en otros, dos.

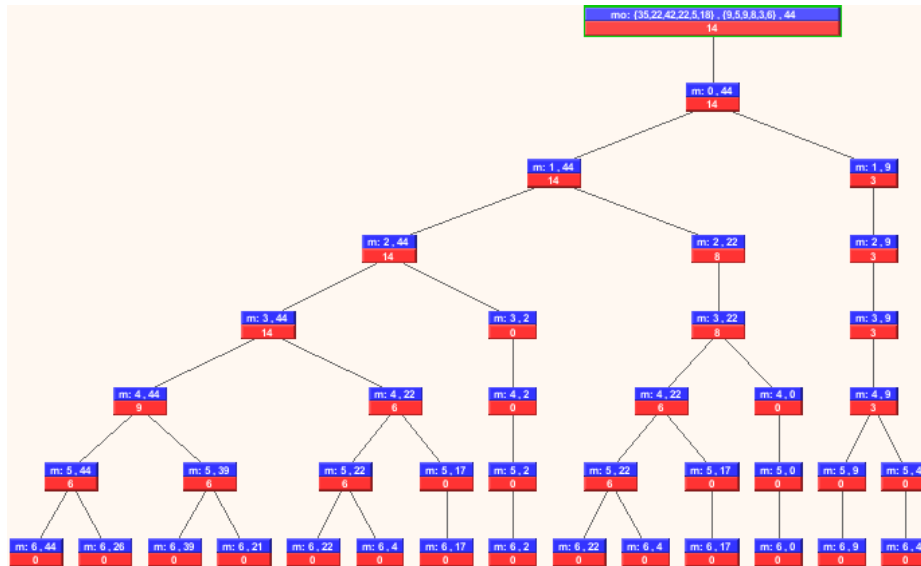


Fig. 2. Árbol de recursión de *mochila01*.

## 2.2 Definición de Familia de Visualizaciones

Nuestra propuesta consiste en que cada algoritmo recursivo múltiple sea explicado mediante un conjunto de árboles de recursión en lugar de uno solo. Dicho conjunto toma la forma de una secuencia de árboles de recursión tales que solamente varía un parámetro (y de forma mínima) entre un árbol de recursión y el siguiente.

Desde un punto de vista práctico, estos algoritmos tienen una complejidad exponencial o factorial, por lo que hay que limitar el número de árboles. Por ejemplo, la Figura 3 contiene el árbol de recursión para *fib(10)*, cuyo gran tamaño ya no permite mostrar el contenido de los nodos. Por tanto, la familia contendrá los árboles de la serie de Fibonacci desde 0 ó 1 hasta un valor menor de 10.

Nuestra hipótesis es que una familia de visualizaciones proporciona una visión más completa de la redundancia del algoritmo, permitiendo superar más fácilmente las dificultades antes señaladas:

- Elección de datos de entrada. Al contener varios árboles de recursión, es muy probable que contenga árboles adecuados para el análisis de redundancia.
- Comprensión del algoritmo. El hecho de que los árboles de recursión contenidos en una familia tengan parámetros sucesivos facilita apreciar el patrón de transición de cada árbol al siguiente. Por tanto, debería ser más fácil



para el alumno comprender las razones de este patrón, es decir, las decisiones en las que se basa el algoritmo.

### 2.3 Clases de Familias de Árboles de Recursión

Aunque todas las familias de árboles de recursión satisfacen la misma definición, las características de cada algoritmo condicionan las decisiones de diseño de cada familia. A continuación damos unas pautas de diseño para los dos casos más importantes que hemos encontrado.

**Familia de Árboles que Dependen del Tamaño de los Datos de Entrada.** Es el caso más sencillo. Se trata de algoritmos cuyo proceso recursivo depende solamente del tamaño de los datos de entrada. Un ejemplo representativo es la serie de Fibonacci, ya que cada activación depende simplemente del valor de su parámetro  $n$ .

Veamos otro problema conocido, el de cambio de monedas [2], en el que se pide el número mínimo de monedas necesario para dar el cambio de cierta cantidad  $c$  de dinero. Es un problema resoluble mediante un algoritmo voraz en ciertos casos pero que en general debe ser resuelto con otras técnicas. El algoritmo recursivo siguiente es la versión inicial para una solución de programación dinámica.

Las unidades monetarias se definen en el array *unidades*, sin más restricciones que la última debe ser unitaria. Por sencillez, las unidades están ordenadas en orden decreciente de valor, desde 200 (la moneda de 2€) hasta 1 (de 1 céntimo).

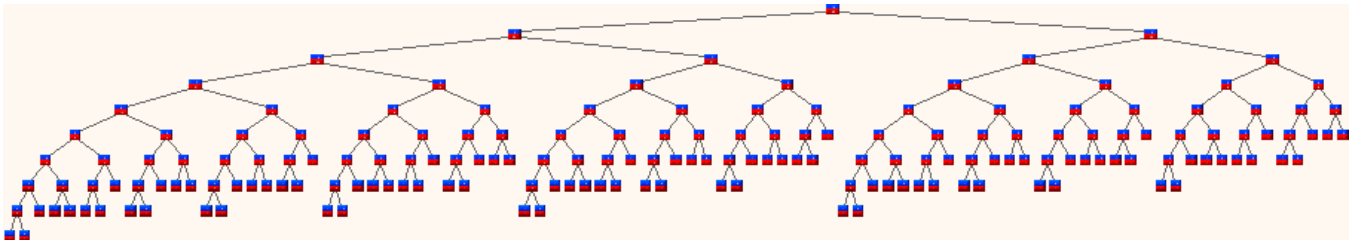


Fig. 3. Árbol de recursión de *fib(10)*.

El método principal llama a un método auxiliar *cm* que va probando, con cada unidad monetaria, todas las formas posibles de cambiar el dinero disponible. Al ser la última unidad monetaria la unidad, las llamadas recursivas del último nivel cambian el dinero  $c$  que queda con  $c$  monedas de un céntimo. Los demás casos corresponden a casos recursivos, donde se prueban todos los cambios posibles para la cantidad  $c$  y la unidad monetaria *unidades<sub>i</sub>* actuales.

```
static int[] unidades =
    {200, 100, 50, 20, 10, 5, 2, 1};
```

```
public static int cambiarMonedas (int c) {
    return cm(0,c);
}

private static int cm (int i, int c) {
    if (i==unidades.length-1)
        return c;
    else {
        int menor = Integer.MAX_VALUE;
        for (int j=0; j<=c/unidades[i]; j++) {
            int result = j + cm(i+1,c-j*unidades[i]);
            if (result < menor)
                menor = result;
        }
        return menor;
    }
}
```

La Figura 4 contiene parte de una familia de árboles de recursión para este algoritmo. Comenzamos con una cantidad  $c$  a desglosar igual a uno y se va aumentando sucesivamente. Por razones de espacio, solamente incluimos árboles de recursión para los valores comprendidos entre 1 y 6, más un árbol más grande, el correspondiente a  $c=14$ . En cada árbol, el nodo superior corresponde a la llamada inicial al método principal y los nodos situados debajo, son llamadas recursivas al método auxiliar  $cm$ , donde cada unidad monetaria se trata en un nivel, desde  $i=0$  (moneda de 2€) hasta  $i=7$  (de 1 céntimo).

Obsérvese que en el árbol más sencillo,  $c=1$ , no puede hacerse el cambio con ninguna unidad monetaria, ya que sólo es posible hacer el cambio con un céntimo. Para  $c=2$ , ya hay dos posibilidades, hacer el cambio con 1 moneda de dos céntimos o con 2 de uno. Para  $c=3$ , las posibilidades para monedas de 2 céntimos siguen siendo las mismas, pero para  $c=4$  ya hay más posibilidades porque puede elegirse entre 0, 1 ó 2 monedas de 2 céntimos.

El árbol correspondiente a  $c=14$  es más complejo porque existen distintas posibilidades para las unidades monetarias de 10, 5 y 2 céntimos. Obsérvese que una familia completa debería contemplar todos los valores de  $c$  hasta poder cambiar al menos una moneda de 2€, es decir, al menos  $c=200$ .

Otros casos muy conocidos de algoritmos de esta clase son:

- Números combinatorios [2]: depende de sus parámetros  $m$  y  $n$ . Obsérvese que en este caso puede definirse una familia bidimensional.
- Alineación de secuencias [5]: depende de los índices que permiten recorrer ambas secuencias, independientemente de su contenido.
- Multiplicación encadenada de matrices [2][3][8]: depende de los índices que delimitan cada subrango de matrices a multiplicar.

- Función de Ackermann: también depende de  $m$  y  $n$ . Hay que tener especial cuidado con esta función porque crece muy rápidamente con valores muy pequeños de  $m$  y  $n$ , por lo que en este caso la familia resulta muy limitada y poco útil.

**Familia de Árboles que Dependen del Tamaño de los Datos de Entrada.** Hay algoritmos cuyo proceso recursivo depende del contenido de los datos de entrada. La diferencia entre esta clase de algoritmos y la anterior es similar a la que se da, en los algoritmos de divide y vencerás, entre la ordenación por mezcla (sólo depende del tamaño del vector) y la rápida (depende también de su contenido).

En esta clase, los árboles de dependencia tienen formas más irregulares, al depender del contenido de los datos de entrada. Algunos ejemplos de algoritmos son:

- Problema de la mochila 0/1 (subsección 2.1.2): el proceso recursivo depende de los pesos de los objetos y de la capacidad de la mochila.
- Subsecuencia común más larga [3]: depende de las dos cadenas de caracteres a comparar.
- Selección de actividades ponderadas [5]: depende de los instantes del solapamiento de las actividades entre sí.

En este caso, es mejor intentar que los árboles de recursión mostrados sean lo menos dispares posible. Para ello, puede fijarse un valor para todos los parámetros menos uno, que es el que varía.

Sea el problema de la mochila 0/1 de nuevo. Los parámetros que controlan el proceso recursivo son los pesos de los objetos y la capacidad libre de la mochila. Para reducir la variación de los árboles, vamos incrementando la capacidad en la unidad. El valor mínimo de la capacidad es el mayor con el que no cabe ningún objeto. Para que la variación se aprecie mejor, los pesos de los objetos solamente difieren en la unidad y se disponen en orden decreciente. El valor máximo de la capacidad será igual a la suma de los pesos de los objetos.

En la Figura 5 incluimos una familia para el problema de la mochila 0/1 con cuatro objetos de pesos  $p=\{5,4,3,2\}$  y beneficios  $b=\{8,1,5,7\}$ . La primera visualización corresponde al caso en que no cabe ningún objeto en la mochila ( $c=1$ ) y sucesivamente hay una posibilidad más que considerar. La última visualización corresponde al caso en que todos los objetos podrían meterse simultáneamente en la mochila ( $c=14$ ).

Puede observarse que los árboles de las dos familias mostradas tienen una estructura similar, muy diferente de la de otros algoritmos recursivos múltiples con redundancia, como la serie de Fibonacci. El parecido de estos árboles de recursión se debe a que sus algoritmos se estructuran en etapas, de forma que en cada etapa se prueba con todas las posibilidades para un candidato. Para el problema de cambio de monedas, los candidatos son las unidades monetarias, pudiéndose escoger un número variable de unidades. Para el problema de la mochila 0/1, los candidatos son los objetos, que sólo pueden introducirse o no en la mochila (opciones respectivas 1 y 0).

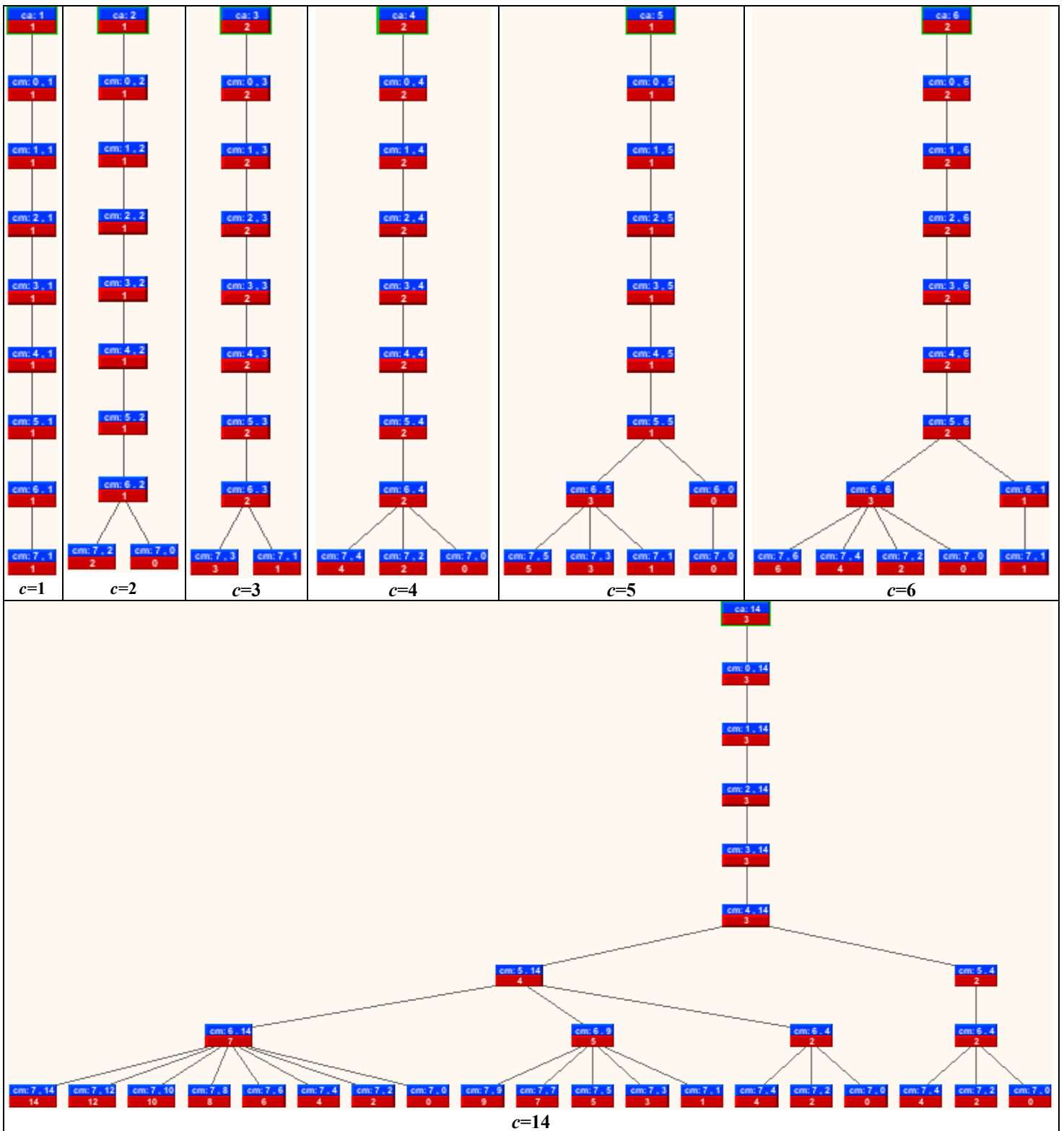


Fig. 4. Parte de una familia de árboles de recursión para el problema de cambio de moneda, con  $c$  de 1 a 6 y  $c=14$ .

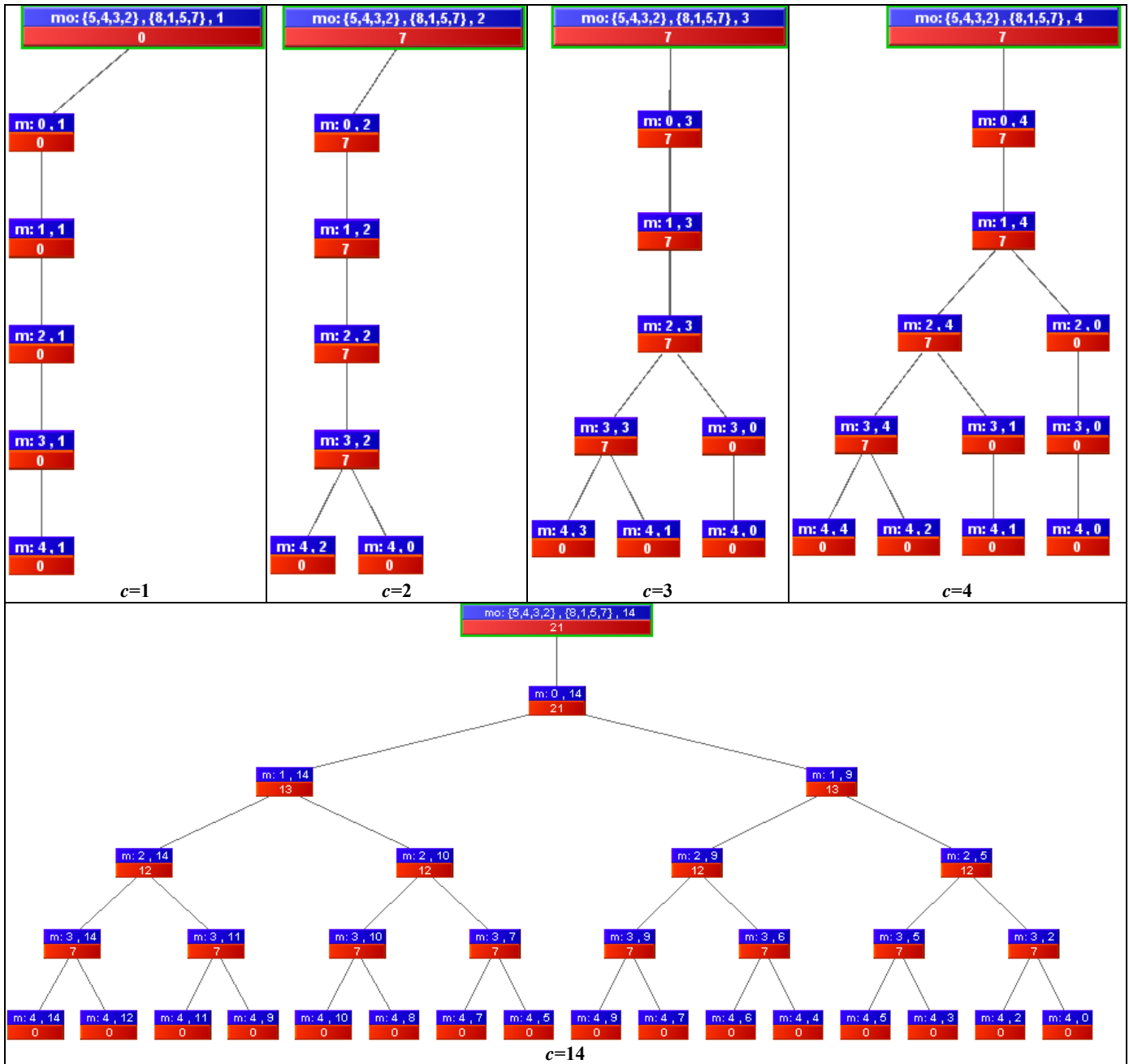


Fig. 5. Parte de una familia de árboles de recursión para el problema de la mochila 0/1, con  $c$  de 1 a 4 y  $c=14$

**Diseño de Familias de Árboles de Recursión.** El número de datos de entrada posibles para un algoritmo suele ser infinito, por lo que podríamos concebir familias de árboles de recursión de tamaño infinito. Sin embargo, nuestro propósito de utilizarlas con fines docentes hace que su tamaño deba ser acotado.

En este apartado esbozamos algunas ideas sobre cómo construir estas familias de forma que sean eficaces educativamente. Las características que deben tener son:

- Número restringido de visualizaciones, a ser posible.
- El tamaño de cada una debe estar acotado, de forma que pueda verse completo y que sean visibles los valores de los parámetros y resultados contenidos en cada árbol.
- La variación de un árbol al siguiente sea mínima.

Ya hemos visto cómo hemos conseguido este efecto en los ejemplos anteriores. Así, en el problema de cambio de moneda, el número de árboles posibles es finito pero demasiado grande (388). Por esta razón, hemos mostrado los 6 árboles más pequeños ( $c \in [1..6]$ ) y otro algo mayor ( $c=14$ ).

En el problema de la mochila 0/1 podemos tener algo más de control. Hemos dejado fijos los valores de los beneficios y hemos variado los pesos de los objetos y la capacidad de la mochila. Con cuatro objetos de  $ps=\{5,4,3,2\}$  y  $c=1$  tenemos un caso simple en el que no puede meterse ningún objeto en la mochila. Aumentando progresivamente la capacidad de uno en uno, se obtienen 14 árboles, cada uno con algunas llamadas recursivas más que el anterior. Es difícil determinar un valor óptimo para cada algoritmo pero debemos descartar familias demasiado numerosas. Obsérvese que con  $ps=\{6,5,4,3,2\}$ , ya necesitaríamos 20 árboles y con  $ps=\{7,6,5,4,3,2\}$ , 27. Por tanto, sería preferible asumir tres objetos de  $ps=\{4,3,2\}$ , resultando 9 árboles, suficientes para observar variabilidad.

En otros casos hay que hacer un análisis más profundo del algoritmo. Así, para la subsecuencia común más larga, podemos tomar dos secuencias de 3 caracteres y probar con secuencias cada vez más distintas. Si fijamos una secuencia, por ejemplo a “000”, y variamos los caracteres de la otra con el carácter ‘1’, podemos obtener 8 casos correspondientes a los números binarios de tres dígitos.

Para el problema de la selección de actividades ponderadas, podemos tomar tres actividades. Analizando las distintas posibilidades de solapamiento, vemos que hay cinco casos distintos y de complejidad creciente.

### 3 Trabajos Relacionados

Es frecuente encontrar una secuencia de visualizaciones para ilustrar el comportamiento de un algoritmo, mostrando la secuencia de estados de su ejecución con ciertos datos de entrada. Sin embargo, una secuencia de visualizaciones del resultado del algoritmo para distintos datos de entrada es atípica.

La situación se comprende mejor si la analizamos en un marco general. En un trabajo anterior [9], propusimos un espacio de visualizaciones con 4 dimensiones: tiempo de ejecución, algoritmo, representación gráfica y datos de entrada. Dicho espacio define un hipercubo de visualizaciones estáticas que, con una interfaz adecuada, permite mostrarlas simultáneamente o animadas. En particular, podría mostrarse una secuencia de visualizaciones en cualquiera de las siguientes dimensiones:

- Tiempo. Corresponde a la usual ejecución de un algoritmo.
- Algoritmo. Corresponde a “múltiples algoritmos”, en que se muestra la ejecución de varios algoritmos simultáneamente.
- Representación gráfica. Corresponde a “múltiples vistas”, en que se muestra la información de una ejecución con varios formatos, normalmente con distintos niveles de abstracción.
- Datos de entrada. Corresponde a una “familia de visualizaciones”, como hemos presentado en este artículo.

Los únicos trabajos que conocemos que encajan en esta última categoría son HalVis [4] y GreedEx [10]. En el primer caso cada algoritmo se ilustra con tres juegos de datos con características muy dispares. En el segundo, el sistema GreedEx muestra los resultados de ejecutar distintos datos, pero en formato tabular (sólo utiliza un formato gráfico para visualizar la ejecución con un conjunto de datos de entrada).

### 4 Conclusiones

Hemos mostrado una forma novedosa de presentar los árboles de recursión. Esperamos obtener mejoras de comprensión y análisis al usarlos con algoritmos con recursividad múltiple redundante. También hemos mostrado cómo usarlos de forma provechosa con distintas clases de algoritmos, según que sus árboles de recursión dependan sólo del

tamaño de sus parámetros o también de su contenido. Por falta de espacio, no hemos podido mostrar otro caso interesante, pero menos importante: mostrar el efecto sobre el proceso recursivo de algunas decisiones de codificación del propio algoritmo recursivo.

Se presentan muchas cuestiones abiertas que merece la pena explorar. Primera, sólo podremos afirmar su utilidad si realizamos una evaluación de eficacia educativa con alumnos; nuestra intención es realizarla en otoño de 2013 en una asignatura de algoritmos avanzados. Segunda, tenemos desarrollada una colección de familias, que podemos ampliar y publicar en la web. Tercera, es interesante estudiar si las familias resultan útiles para otras clases de algoritmos, p.ej. de divide y vencerás. Cuarta, el sistema SRec permite exportar visualizaciones generadas durante una ejecución. Para soportar la creación de familias de visualizaciones, debería pedir al usuario un rango de parámetros y que generara automáticamente la familia a partir de la visualización del estado final de cada ejecución. Por último, sería útil evaluar si una familia es más útil presentada como una secuencia de visualizaciones estáticas o en forma de animación (incluso con transiciones continuas con efecto de *morphing*).

**Agradecimientos.** Este trabajo se ha financiado con el proyecto TIN2011-29542-C02-01 del Ministerio de Economía y Competitividad.

## Referencias

- [1] Bird, R. S.: 1980. Tabulation techniques for recursive programs. *ACM Computing Surveys* 12, 4 (diciembre 1980) 403-417. DOI= 10.1145/356827.456831
- [2] Brassard, G. and Bratley, P.: *Fundamentals of Algorithmics*. Prentice-Hall (1996)
- [3] Cormen, T. H., Leiserson, C. E., Rivest, R. L., Stein, C.: *Introduction to algorithms*, 3ª ed. The MIT Press, Cambridge, MA, EE.UU. (2009)
- [4] Hansen, S., Schrimpscher, D., Narayanan, N.H.: Designing educationally effective algorithm animations. *Journal of Visual Languages and Computing* 13, 3 (2002) 291-317. DOI= 10.1006/jvlc.2002.0236
- [5] Kleinberg, J., Tardos, É.: *Algorithm Design*. Pearson Addison-Wesley (2006)
- [6] Pettorossi, A.: A powerful strategy for deriving efficient programs by transformation. In *Proceedings of the ACM Symposium on Lisp and Functional Programming*. LFP'84. ACM, New York, NY, EE.UU., 273-281. DOI= 10.1145/800055.802044



- 
- [7] Price, B., Baecker, R., Small, I.: An introduction to software visualization. In *Software Visualization*, Stasko, J., Domingue, J., Brown, M. H. and Price, B. A. (eds). The MIT Press, Cambridge, MA, EE.UU., 3-27 (1998)
- [8] Sahni, S.: *Data Structures, Algorithms and Applications in Java*. Silicon Press, Summit, NJ, EE.UU. (2005)
- [9] Velázquez-Iturbide, J. Á.: Characterizing time and interaction in a space of software visualizations. In *Proceedings of the 6<sup>th</sup> Program Visualization Workshop (Darmstadt, Alemania, 30 junio 2011)*. PVW'11. Technical Report TUD-CS-2011-0153, Technische Universität Darmstadt, Germany, 43-51
- [10] Velázquez-Iturbide, J. Á., Debdi, O., Esteban-Sánchez, N., Pizarro, C.: GreedEx: A visualization tool for experimentation and discovery learning of greedy algorithms. *IEEE Transactions on Learning Technologies* 6, 2 (Abril-Junio 2013), 130-143. DOI= 10.1109/TLT.2013.8
- [11] Velázquez-Iturbide, J. Á., Pérez-Carrasco, A.: InfoVis interaction techniques in animation of recursive programs. *Algorithms* 3, 1 (Marzo 2010), 76-91. DOI= 10.3390/a3010076
- [12] Velázquez-Iturbide, J.Á., Pérez-Carrasco, A., Urquiza-Fuentes, J.: SRec: An animation system of recursion for algorithm courses. In *Proceedings of the 13th Annual Conference Innovation and Technology in Computer Science Education (Madrid, Spain, June 2008)*. ITiCSE'08. ACM, New York, NY, EE.UU., 225-229. DOI= 10.1145/1384271.1384332

## El intercambio de papeles como estrategia docente

Isidoro Hernán Losada

<sup>1</sup> Departamento de Lenguajes y Sistemas Informáticos I,  
Universidad Rey Juan Carlos  
28933 Móstoles, Madrid, Spain  
isidoro.hernan@urjc.es

**Abstract.** One of bigger challenges in the university teaching is how promoting the pupils' active and significant learning. In this article we presented a learning experience using the role-playing technique. The student must switch his role as receptor of information and becoming an active learner. On the one hand, he has to synthesize and to transfer the knowledge to his peers (teacher's role). On the other hand, he is a reviewer of information produced by partners (reviewer's role). Teacher's role was performed in a group activity, promoting at the same time collaborative work, and the reviewer's role was accomplished in an individual task. The analysis of data collected demonstrates its validity as motivating and efficient technique in the learning process and students reported better understanding of the concepts using the role-playing.

**Keywords:** Role-playing, active learning, teaching techniques, learning process.

### 1 Introducción

Uno de los mayores retos a los que se enfrenta un docente en la enseñanza universitaria es cómo fomentar el aprendizaje activo y significativo de los alumnos. Con la implantación de los nuevos estudios adaptados al Espacio Europeo de Educación Superior (EEES) [1] para favorecer, en materia de educación, la convergencia europea, surgen nuevos retos. Los planes de estudio de los grados en Ingeniería del Software y en Ingeniería de Computadores definen sus asignaturas en base al tiempo que el alumno ha de emplear para conseguir las competencias adecuadas. Este es el motivo que sitúa al alumno como actor principal de su proceso de aprendizaje. El nuevo

paradigma de la educación conlleva la necesidad de adaptar las metodologías docentes de forma que fomenten al estudiante a tomar un papel activo, mejorando su iniciativa y su pensamiento crítico. Existen varias teorías pedagógicas relacionadas con el aprendizaje activo. La teoría constructivista y la sociocultural son las corrientes más influyentes en este tipo de aprendizaje.

La idea principal del constructivismo [2] es que el conocimiento no es un objeto estático sino que es un ente en continua evolución. El alumno construye su propio conocimiento a través de sus experiencias. El conocimiento previo sirve de soporte al conocimiento nuevo. La información nueva recibida es asimilada y añadida a su red de conocimientos y experiencias previas. Por lo tanto, esto implica que el aprendizaje es un proceso personal que cada individuo va transformando de forma continua dependiendo de sus propias vivencias. Los discentes tienen que asumir su compromiso en este proceso y desarrollar sus habilidades y capacidades para guiar su propio conocimiento y sus competencias. El docente asume su papel de guía del aprendizaje y deja de ser el elemento más importante en el proceso de enseñanza-aprendizaje.

La teoría sociocultural [3] destaca que el aprendizaje significativo solo se logra en un contexto social. Formula que la inteligencia humana emana de nuestra cultura o de nuestra sociedad, y que el aprendizaje individual ocurre de forma predominante debido a nuestra interacción con el entorno social [4]. Los estudiantes transforman el conocimiento en conceptos. Estos conceptos pueden irse relacionando con otros adquiridos previamente. El aprendizaje se produce de forma principal como resultado de analizar y comprender diferentes alternativas a través del diálogo con otros compañeros. Cuando las personas trabajan colaborativamente [5] en una actividad pueden percibir la tarea desde diferentes puntos de vista y son capaces de solucionar problemas mediante la negociación o discusión. Estas soluciones son conseguidas gracias al conocimiento compartido por todos los miembros del equipo. En este marco el docente debe ser un mero intermediario con la cultura social. Su trabajo es organizar el aula de forma que ayude a promover las interacciones, la creación de expectativas y a generar un clima de confianza. De esta forma, a través de actividades conjuntas e interactivas, el alumno genera su conocimiento, en parte gracias a sus aportes y en parte gracias a las ayudas dadas en las actividades propuestas. Estas ayudas deben ser expuestas de tal forma que guíen al discente a alcanzar el objetivo marcado.

Para fomentar los dos tipos de aprendizaje (activo y colaborativo) se ha utilizado el intercambio de roles o *role-playing*. El *role-playing* se define [6] como “una experiencia en la se pide al participante que actúe como si fuera una persona particular en una situación particular”. En ella, el alumno adopta un rol específico en un contexto marcado, busca información sobre la nueva situación. Al recibir esta información surgen dudas y problemas sobre los que tendrá que reflexionar. Después interacciona y debate con sus compañeros lo que puede conllevar una nueva búsqueda de información o una solución a las dudas y problemas. La etapa final es la de reflexión sobre todo el proceso y el posterior aprendizaje significativo. Esta técnica ha sido utilizada tradicionalmente en ejercicios de mejora de habilidades sociales, pero también ha sido usada en otros contextos [7], como por ejemplo, en la enseñanza de la programación orientada a objetos [8].

La utilización de este método haciendo al alumno asumir los papeles de profesor y/o evaluador de sus compañeros ofrece una serie de ventajas entre las que se pueden destacar las siguientes:

- El alumno debe ser creativo, ya que asumir otro rol requiere imaginar la nueva situación y el comportamiento del nuevo personaje, por lo que la tarea puede verse más como un juego que como un trabajo.
- Aumenta la autocrítica. La nueva situación hace que el participante deba reflexionar sobre su papel real.
- Ofrece la sensación de ser ‘dueño’ del proceso de aprendizaje. El alumno es el encargado de buscar, organizar y resumir la información que ha de aprender.
- Tiene que ver la forma de convertir el conocimiento implícito en explícito. Explicar un concepto hace visibles los conocimientos previos y estos han de ser mostrados al resto de compañeros para una mejor comprensión. Este proceso mejora el aprendizaje significativo.
- Tiene la oportunidad de aprender y reflexionar sobre los diferentes puntos de vista y roles en el proceso de enseñanza/aprendizaje.
- Incrementa la empatía y tolerancia. Situarse en el lugar de otra persona hace ver su punto de vista y mirarse a uno mismo desde el exterior.
- La presión de tener que explicar el tema a los compañeros motiva más que tener que aprender para el examen. De esta forma el aprendizaje pasa de ser temporal a ser duradero.

El resto de la contribución se estructura como sigue. En la sección II se describe la experiencia realizada dividida en dos casos de estudio, incluyendo el contexto en el cual se realizaron. A continuación, en la sección III mostramos los datos obtenidos de las distintas actividades y se hace el análisis de los mismos. Finalmente, en la sección IV describimos las conclusiones de nuestro trabajo.

## **2- Descripción de la experiencia**

### **Contexto**

La técnica del *role-playing* fue aplicada en la asignatura de Programación Orientada a Objetos, de los grados de Ingeniería del Software (GIS) e Ingeniería de Computadores (GIC) de la Universidad Rey Juan Carlos. Esta materia se encuentra en el segundo curso y se imparte en el primer cuatrimestre. Consta de 6 créditos ECTS (*European Credit Transfer System*) y es de carácter obligatorio. Los alumnos acceden a este segundo curso, habiendo recibido las asignaturas de Introducción a la Programación, donde se enseña programación imperativa, y Estructuras de Datos, donde se imparten conocimientos de estructuras de datos básicas como por ejemplo pilas o colas, entre otras.

La investigación se ha llevado a cabo realizando dos actividades de aprendizaje distintas. En la primera de ellas los alumnos adoptaban el papel del profesor, preparando un determinado tema para dar una clase magistral a sus compañeros. En el segundo, los discentes tomaron el papel de evaluador del trabajo producido por sus compañeros de un tema sobre el cual tenían conocimientos previos. A continuación pasamos a detallar cada actividad.

### **Rol de profesor**

En este caso de estudio, se dividió la clase en grupos de trabajo de 2/3 personas. Cada grupo eligió un tema de una lista de 26 trabajos propuestos por el equipo docente, de forma que no hubiera temas repetidos dentro de alumnos del mismo grado. Los temas estaban relacionados con aspectos avanzados de la asignatura [9]. La asignación del tema fue por orden cronológico de petición. El trabajo consistió en preparar una

presentación sobre la materia, que debían contener obligatoriamente los siguientes apartados:

- a) Una introducción con aspectos generales, características más importantes, ventajas e inconvenientes.
- b) Ejemplos sencillos que ilustraran el tema tratado y que facilitaran su comprensión.
- c) Un ejercicio simple que resolverían sus compañeros en clase con la información recibida.
- d) La solución al ejercicio anterior.
- e) Bibliografía utilizada en la preparación del trabajo.

Cuando los alumnos terminaron de preparar cada uno de los temas, el portavoz de cada grupo fue el encargado de enviar la presentación al equipo docente. Al finalizar el plazo de entrega, se fijó un día de presentación de trabajos. En esa jornada se expusieron públicamente todos los trabajos. La tarea de explicar el tema le fue asignada a un alumno del grupo al azar. Con este criterio de exposición aleatoria, lo que se quería asegurar por parte del equipo docente era conseguir que todos los alumnos se implicaran activamente a la hora de preparar el tema asignado. Este criterio era conocido por los alumnos antes de comenzar a preparar el trabajo. Al fin de la presentación, hubo un turno de preguntas y dudas sobre el tema expuesto donde podían intervenir todos los alumnos de la clase y los profesores de la asignatura, que en este caso, actuaban como alumnos.

El equipo docente evaluó tanto el trabajo (70% de la nota) como la presentación realizada (30% de la nota), siendo los criterios de puntuación públicos y conocidos previamente al desarrollo del trabajo.

La realización de esta actividad estaba descrita en la guía docente de la asignatura y calificada como evaluable, con un porcentaje del 10% de la nota final del alumno, por lo que sin llegar a ser obligatoria para aprobar la asignatura la participación fue de 116 alumnos sobre 126 matriculados (un 92.06%)

Se entregaron y expusieron un total de 42 trabajos, que coincidió con el número de grupos formados en ambos grados (en GIC 23 grupos y en GIS 19 grupos), por lo que no hubo ningún abandono durante la experiencia.

### **Rol de Evaluador**

Al contrario que el trabajo anterior, esta actividad fue catalogada como voluntaria e individual y premiada con 0.3 puntos adicionales a la nota final. El trabajo consistió en valorar las transparencias del tema elegido por el grupo del alumno pero generadas por otro grupo de un grado distinto. Los discentes debían valorar con una puntuación de 0 a 10 (un 0 denota que no han realizado ese apartado y un 10 que lo tienen muy bien) cada uno de los siguientes puntos: Introducción al tema seleccionado (por ejemplo presentando aspectos generales, características más importantes, ventajas, inconvenientes, etc.); los ejemplos sencillos sobre los conceptos introducidos en el tema seleccionado; el ejercicio que realizarán el resto de los estudiantes cuando se realice la exposición del tema; la solución al ejercicio anterior; la bibliografía utilizada; la claridad de las transparencias; la información correctamente estructurada; la profundidad con que se trata el tema; la adecuación al nivel de los alumnos (segundo curso del grado, con conocimientos de POO); y la valoración global, siendo este último apartado el que se ha tenido en cuenta para poner la nota adicional al alumno.

El motivo de evaluar un trabajo sobre el mismo tema preparado es disminuir la dificultad de realizar esta tarea, ya que al tener un mayor conocimiento de la materia, el alumno se encuentra más capacitado para juzgar de una forma crítica y constructiva el trabajo de sus compañeros según unos criterios de valoración específicos facilitados por el equipo docente. Los 0.3 puntos adicionales a la nota se conseguían si la valoración entregada coincidía con la evaluación realizada por el equipo docente. Cada punto de diferencia entre ambas valoraciones restaba 0.1. En este sentido, se premiaba que los alumnos evaluaran críticamente el trabajo de sus compañeros ajustándose lo máximo posible al criterio del profesor.

El número de alumnos que realizaron esta actividad voluntaria fue 56, donde solo hubo una baja durante el proceso.

Al finalizar esta actividad, los participantes debieron rellenar una encuesta de opinión sobre las dos actividades de aprendizaje realizadas cuyos resultados se presentan en la siguiente sección.

### 3. Datos y análisis

En esta sección se muestran los datos obtenidos en ambos casos de estudio y se hace un análisis de los resultados.

#### Rol de profesor

Como se informó en el apartado anterior, se entregaron y expusieron públicamente 42 trabajos grupales, que involucraron a 116 alumnos de 126 matriculados. Solo 1 grupo constituido por tres alumnos suspendió esta experiencia (nota menor que 5). Las notas de esta actividad se muestran en la figura 1. Hubo 15 sobresalientes (nota mayor o igual a 9), 65 notables (nota mayor o igual que 7 y menor que 9), y 23 aprobados (nota mayor o igual a 5 y menor que 7). Esto significa que superaron la actividad el 97.41% de los alumnos que prepararon y presentaron el trabajo. De los 113 alumnos que aprobaron esta actividad 54 superaron la asignatura de programación orientada a objetos. Esto significa que el 47.78% de los alumnos que hicieron correctamente el trabajo aprobaron la asignatura completa.

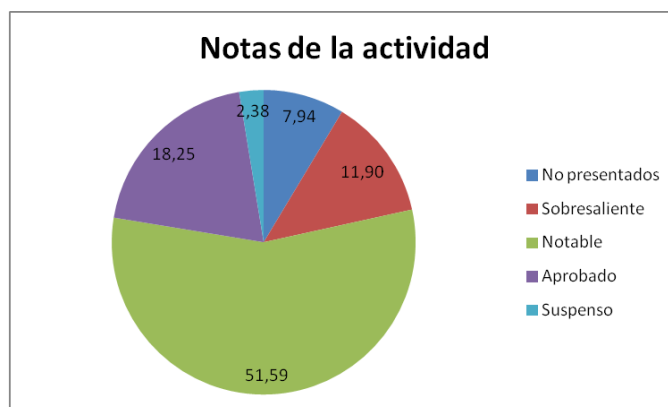
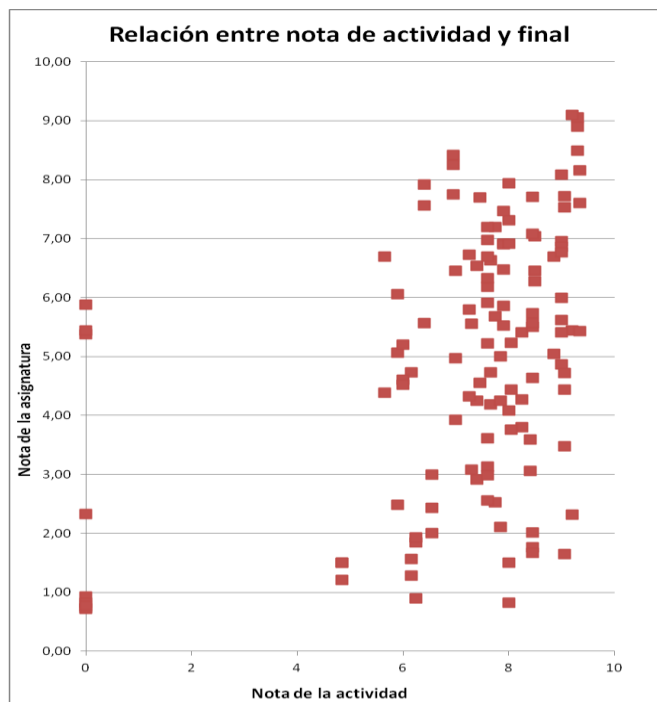


Fig. 1. Distribución de los porcentajes de las notas obtenidas en la actividad de rol de profesor.

Se ha realizado un estudio sobre la correlación entre la nota obtenida en esta actividad y la nota final de la asignatura. El coeficiente de correlación lineal de Pearson da un valor  $r = 0.44$ . La figura 2 muestra el gráfico de dispersión de ambas notas. Del valor de  $r$  y de la observación de la gráfica se puede deducir que no existe una relación directa entre el desempeño de esta actividad y la nota final de la asignatura. Esta conclusión es lógica, ya que al ser una actividad grupal, no todos los miembros del



equipo trabajan por igual, aunque la nota de la actividad es la misma para todos los integrantes. Hay que observar que el coeficiente de correlación de Pearson está fuertemente influenciado por valores extremos. En la figura 2 se observa que hay varios puntos sobre el eje Y debido a los 10 alumnos que no realizaron la actividad.



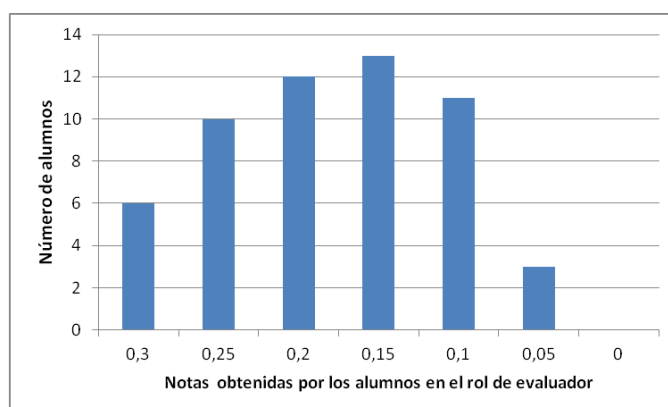
**Fig. 2.** Gráfica de dispersión entre la nota final de la asignatura y la nota obtenida en la actividad tomando el rol de profesor.

Por otra parte, el porcentaje de alumnos que han realizado el trabajo y que han aprobado en la asignatura es del 47,78% frente al 42,85% de alumnos matriculados que han aprobado. La diferencia es pequeña y puede ser debida al carácter obligatorio de la prueba.

### **Rol de evaluador**

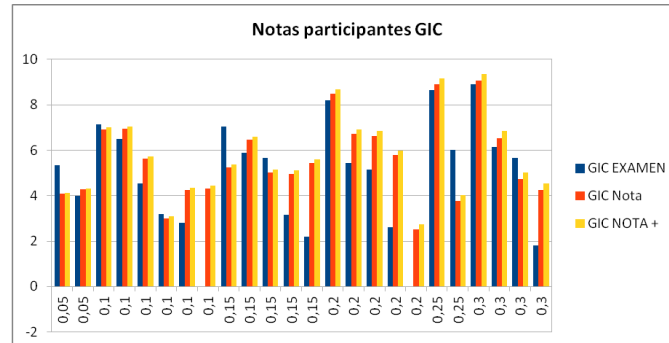
En la actividad de evaluación del trabajo realizado por compañeros de otros grados, participaron 55 alumnos. La figura 3 muestra las notas obtenidas por los discentes en esta actividad. Los participantes obtuvieron una nota de 0,3 puntos si su valoración

coincidió con la del equipo docente. Por cada punto de diferencia entre ambas evaluaciones, se restaba 0,1 puntos.



**Fig. 3.** Gráfica de barras con las valoraciones de los alumnos de la actividad del rol de evaluador.

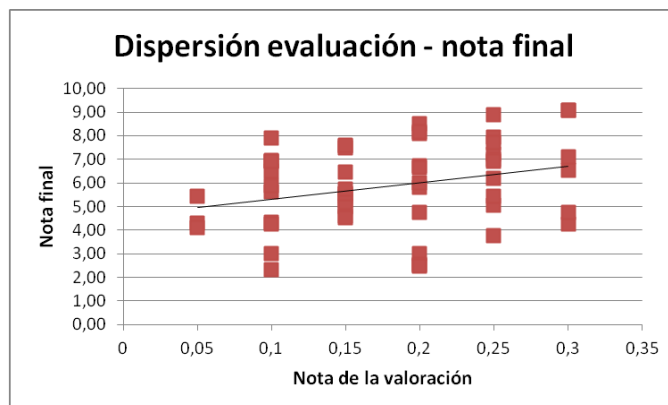
Hay que reseñar que el 51% de los alumnos tuvo una diferencia de nota igual o inferior a 1 punto respecto al criterio del profesorado (6 alumnos coincidieron con la valoración del equipo docente, 10 alumnos tuvieron una diferencia de 0.5 puntos y 12 sobre la nota final de la asignatura, se realizó un estudio más detallado sobre las notas de los alumnos que realizaron esta actividad. En la figura 4 se muestra un gráfico de barras de las notas obtenidas por los 24 alumnos de GIC (por claridad y al obtenerse resultados análogos en el grupo de GIS, sólo se presenta una de las gráficas). El eje x representa la nota de esta actividad y el eje y las notas obtenidas en el examen final (40% de la nota final), en la nota final y en la nota final más los puntos adicionales obtenidos por la realización de esta actividad.



**Fig. 4.** Gráfico de barras donde se muestra la relación entre la nota obtenida en la actividad tomando el rol de evaluador (eje x) y las notas obtenidas en el examen, la nota final y la nota final más la nota de la actividad.

Se puede intuir un ascenso progresivo de la altura de las barras (notas de examen y finales) a medida que crece la nota obtenida en la actividad, lo cual nos puede inducir a pensar que los alumnos que obtienen más nota en esta tarea voluntaria, sacan unos mejores resultados en su nota final. Por este motivo, se realizó un estudio de la correlación entre la nota de la actividad y la nota del examen, obteniéndose un valor de  $r = 0.25$ . El valor del coeficiente de Pearson para la relación entre la nota de la actividad y la nota final es de  $r = 0.27$ . En ambos casos, el valor es muy bajo, lo que conlleva a concluir que no existe una relación fuerte y la relación entre ambas es directa (dado el valor positivo de  $r$ ).

La figura 5 muestra el gráfico de dispersión entre la nota obtenida con la actividad tomando el rol de evaluador y la nota final de la asignatura. En esta figura, se ha añadido la línea de tendencia lineal, que corrobora la idea intuitiva que una mayor nota en la actividad va ligada a una mejor nota en la asignatura.



**Fig. 5.** Gráfica de dispersión entre la nota final de la asignatura y la nota obtenida en la actividad tomando el rol de evaluador. Se ha añadido la línea de tendencia.

### Encuesta de opinión

Los alumnos participantes en la actividad anterior rellenaron una encuesta de opinión sobre las dos actividades realizadas descritas en los dos apartados anteriores. Se obtuvieron un total de 55 respuestas. Cada uno de los apartados se podía puntuar de 0 a 10, donde un 0 significa que está en total desacuerdo o que considera que no es nada interesante y un 10 que se encuentra totalmente de acuerdo o que le parece muy interesante, salvo el último, que era un apartado para recoger comentarios sobre ambas actividades. Las preguntas formuladas fueron las siguientes:

- P1: La actividad de tomar el rol de profesor ¿te parece útil para la asignatura de programación orientada a objetos?
- P2: ¿Te ha aportado mucho a tu aprendizaje?
- P3: ¿Consideras que la preparación de los trabajos es una buena iniciativa para la asignatura?
- P4: La revisión de otro trabajo similar al tuyo ¿te aporta mucho?
- P5: Al revisar el trabajo de tus compañeros ¿Te has puesto en el papel del profesor?
- P6: ¿Te parece interesante cambiar de rol (de alumno a profesor/evaluador)?
- P7: ¿Crees que evaluar a tus compañeros favorece el aprendizaje del tema?
- P8: ¿Crees que realizar un trabajo sobre un tema concreto favorece el aprendizaje del tema?
- P9: La presentación del trabajo en clase ¿te parece interesante?
- P10: Si tienes algunas sugerencias o mejoras sobre las actividades realizadas, por

favor, escríbelas aquí.

La Tabla 1 muestra los estadísticos más representativos extraídos de los resultados a las preguntas anteriores. Realizando un análisis de los resultados de esta encuesta podemos afirmar que los aspectos mejor valorados de estos casos de estudio son la pregunta P8 (la preparación del tema favorece su aprendizaje; media 8.41 y mediana 9), la pregunta P5 (al revisar el trabajo de mis compañeros he adoptado el papel del profesor; media 7.74 y mediana 8) y la pregunta P1 (el cambiar al rol de profesor les parece una actividad útil; media 7.62 y mediana 8). Estas opiniones hacen que la primera actividad haya sido muy bien valorada por los alumnos, considerando

**Tabla 1.** Resultados obtenidos en la encuesta de opinión.

Pregunta	Media	Mediana	Desviación típica
P1	7.62	8	1.39
P2	6.82	7	1.89
P3	7.55	8	1.63
P4	6.89	7	1.93
P5	7.74	8	1.94
P6	7.47	8	1.75
P7	6.39	7	2.29
P8	8.41	9	1.67
P9	6.69	7	2.21

Los aspectos peor valorados de los encuestados corresponden a las preguntas P7 (la evaluación de un trabajo similar favorece el aprendizaje; media 6,39 y mediana 7) y P9 (la presentación del trabajo en clase te parece interesante; media 6,69 y mediana 7). Los alumnos consideran que la valoración de otro trabajo sobre el mismo tema no favorece de manera especial su aprendizaje. Quizás sea debido a que sólo se les exigía poner una nota sin tener que justificar por escrito dicha valoración. Sin esta argumentación, la evaluación se debió hacer sin necesidad de reflexionar sobre el tema, por lo que no les promovió el aprendizaje. La exposición pública del tema en clase no una práctica habitual en estos estudios, por lo que les cuesta mucho enfrentarse a este reto. De hecho, es la primera vez a lo largo del grado que los alumnos se tienen que enfrentar a

una exposición pública. Por este motivo y por no saber con antelación quién era el responsable de la presentación, se puede justificar esa valoración más baja. La desviación típica de ese apartado es de las mayores, lo que indica un alto grado de disparidad en las opiniones de los alumnos, seguramente provocadas por las distintas personalidades y grados de extroversión.

En la encuesta también se introdujo un apartado de sugerencias para que los alumnos pudieran expresar de una forma más detallada su opinión respecto a la realización de estas dos actividades. Entre los comentarios recibidos, podemos destacar los siguientes para el rol de profesor: *“Es mejor una exposición pública grupal que individual”*; *“el tiempo de exposición es corto”*; *“En caso de no poder hacer la exposición grupal, estaría bien conocer el componente del grupo que la hará con antelación”*; *“las actividades han durado poco tiempo corto y se han realizado en fechas cercanas a exámenes”*; *“en general las presentaciones de otros compañeros suscitan poco interés y estaría mejor si las presentaciones se hacen a lo largo del curso”*.

Como se puede observar, la mayoría de los comentarios están relacionados con algún aspecto de la presentación pública del trabajo, actividad que se enfrentaban por primera vez. Respecto a los comentarios sobre la fecha de realización del trabajo, esta actividad tiene lugar al final del cuatrimestre ya que requiere un cierto grado de madurez por parte de los alumnos en la materia de programación orientada a objetos para conseguir abordar con éxito los temas planteados.

En los comentarios sobre el rol de evaluador destaca “la dificultad de evaluar el contenido sin asistir a la presentación”. Hay que recordar el hecho que los alumnos debían valorar el trabajo entregado por compañeros de otra titulación, por lo que no habían asistido a la jornada de presentación pública.

#### **4. Conclusiones**

En este artículo se ha presentado una experiencia donde los alumnos participaban en dos actividades de aprendizaje donde se utilizaba la técnica de *role-playing*. Esta experiencia fue realizada durante el último mes de clase y constó de dos cambios de papel por parte de los discentes. El primero fue asumir el rol de profesor, preparando y explicando un tema a los compañeros durante una exposición pública. El segundo fue adoptar el rol de evaluador valorando un trabajo afín al tema que se había preparado en

la actividad previa.

En la actividad en la que el alumno toma el rol de profesor, los análisis de los datos muestran que no existe una fuerte dependencia entre las notas obtenidas en la actividad y las notas finales en la asignatura. Esto puede ser debido a que la calificación de la experiencia es grupal.

En la actividad en la que el alumno toma el rol de evaluador se puede observar la relación entre la nota adicional, conseguida al haber coincidido con el evaluación del experto, con la nota final obtenida en la asignatura. La participación voluntaria en esta experiencia puede llevar a indicios sobre la realización real del trabajo, ya que la evaluación de un tema que ha sido preparado en menos costosa (y más eficaz) que realizarla sobre un tema novedoso o poco conocido.

También podemos afirmar que los alumnos o alumnas que deciden hacer una actividad voluntaria (rol de evaluador), tienen un mayor porcentaje de aprobados (56,36%) que los que hacen una actividad obligatoria (rol de profesor 47.78%). De hecho la actividad de tomar el papel de profesor fue mejor valorada que la de evaluador entre los estudiantes.

Las encuestas de opinión muestran la satisfacción de los alumnos ante esta técnica novedosa para ellos e instan a la utilización de la misma en otras asignaturas. Esto nos anima a incluirla en los años venideros como una técnica propia con la que trabajar distintas actividades de aprendizaje en la materia.

**Agradecimientos:** Este trabajo se ha financiado con el proyecto TIN2011-29542-C02-01 del Ministerio de Innovación y Ciencia.

## Referencias

- [1] Espacio Europeo de Educación Superior. Disponible en: <http://www.eees.es/>. Consultada el 25/11/2012.
- [2] J. Bruner, *Toward a Theory of Instruction*. Cambridge, MA: Harvard University Press. 1966.
- [3] L.S. Vygotsky, *Mind in society: The development of higher psychological processes*. Cambridge MA: Harvard University Press. 1978
- [4] D.W., Johnson and F.P., Johnson, *Learning Together: Group Theory and Group Skills*. Pearson Education, 10th Edition, 2009.
- [5] D.W. Johnson, R.T. Johnson, and K. Smith, *Cooperative learning: Increasing college faculty instructional productivity*. (ASHEERIC Higher Education Report No. 4). Washington, DC: The George Washington University, School of Education and Human Development. 1991

- 
- [6] E. Aronson, and J. M. Carlsmith, J. M. Experimentation in social psychology. The handbook of social psychology, 2(2), 1968 pp. 1-79.
- [7] T. R. Henry , J. LaFrance, Integrating role-play into software engineering courses, Journal of Computing Sciences in Colleges, v.22 n.2, p.32-38, December 2006
- [8] G. Jiménez-Díaz, M. Gómez-Albarrán, M. A. Gómez-Martín and P. A. González-Calero, “Understanding Object-Oriented Software through Virtual Role-Play”, 5th IEEE International Conference on Advanced Learning Technologies, Kaohsiung, Taiwan, IEEE Computer Society, 2005.
- [9] Martín, E., Lázaro, C, Hernán-Losada, I.: Active Learning in Telecommunication Engineering: A case study. Proceedings of IEEE Engineering Education 2010 – The Future of Global Learning in Engineering Education, pp. 374 (Abstract), ISBN 978-1-4244-6569-9 (full paper). Madrid, Spain. 14-16 Abril 2010.



# Kinect con Intrael

Francisco Domínguez Mateos

Departamento de Lenguajes y Sistemas Informáticos I,  
Universidad Rey Juan Carlos, Madrid, Spain  
francisco.dominguez@urjc.es

**Abstract.** En este trabajo se pretende exponer una tecnología desarrollada para facilitar el trabajo con Kinect, reduciendo todas las tareas relacionadas con la gestión del periférico, obtención de información, filtrado y segmentación a simples peticiones de servicios web con JSON a través del protocolo HTTP.

**Keywords:** interacción persona ordenador, visión por computadora, JSON, servicios web, HTTP.

## 1 Introducción

En este trabajo se pretende exponer una tecnología desarrollada basada en un software denominado Intrael, ideada para facilitar el trabajo de interacción con Kinect, reduciendo todas las tareas relacionadas con la gestión del periférico, obtención de información, filtrado y segmentación a simples peticiones de servicios web con JSON a través del protocolo http.

Con el uso normal de Kinect, es necesario interactuar con los drivers del mismo a través de una compleja arquitectura de APIs. Actualmente existen dos versiones de drivers, con sus correspondientes APIs, los propios de Microsoft y los de Primesense. Este último es el autor real del sistema interno de Kinect, a quien Microsoft compró los derechos para Kinect.

Intrael nos abstrae completamente la API y nos proporciona el preprocesamiento y segmentación adecuados para identificar elementos individuales dentro de la escena que se está observando. Además permite acceder a dicha información de una forma muy sencilla, abierta y distribuica ya que utiliza una versión de servicios web basados en el protocolo http y la tecnología JSON.

En el presente trabajo se introduciremos los conceptos básicos de Kinect y cual es el fundamento de su funcionamiento, a continuación comentaremos el software Intrael, su arquitectura, como configurarlo y utilizarlo, bien que resulta tremendamente sencillo. Para ilustrar la sencillez de Intrael, veremos un ejemplo realizado en Python, que con tan solo tres líneas de código, nos proporciona información sobre los objetos identificados en la escena capturada por Kinect. Finalmente mostraremos algunos proyectos en los que se está utilizando esta tecnología.

## 2 ¿Que es Kinect?

Kinect es el sensor de la videoconsola Xbox de Microsoft, permite interactuar con el sistema sin que el usuario porte ningún tipo de dispositivo.

Realmente es un escáner 3D en tiempo real, junto con un software de inteligencia artificial que es capaz de reconocer los elementos dentro de la escena que observa y las posturas del cuerpo humano, de las personas que hay dentro de la escena.

Está compuesto principalmente por una cámara de video a color con una resolución de 640x480 y sensor CMOS, junto con el sensor de profundidad que es, otra cámara de infrarrojos con resolución 640x480 y sensor CMOS, acompañada con un proyector de puntos estructurados de luz infrarroja.

También dispone de un array de micrófonos ideados para poder realizar reconocimiento de voz y un motor que permite controlar el cabeceo de Kinect, con lo que se puede ampliar el campo de visión o actuación del sistema.



**Fig. 6.** En esta figura se ilustra donde se encuentran distribuidos cada uno de los elementos que constituyen el dispositivo Kinect.

Junto con el dispositivo, es necesario el software que están formados por los drivers junto con la API del SDK, que permite su utilización y añade la funcionalidad adicional relacionada con inteligencia artificial, reconocimiento de elementos y posturas. Existen dos versiones de drivers y API, una es de Microsoft y la otra la empresa Primesense, propietaria de la patente.

### 3 Como funciona Kinect

La funcionalidad de Kinect viene dividida en dos partes, la aportada por el propio dispositivo junto con los drivers y la que proporciona el software asociado al dispositivo para reconocimiento de formas y posturas. Nos vamos a centrar en la primera, donde básicamente se trata de un escáner 3D en tiempo real. Para llevarlo a cabo el dispositivo dispone de un potente procesador interno que se encargará de realizar las conversiones de la luz estructurada recogida en la cámara infrarrojos y convertirla en información de profundidad o distancia a los objetos de la escena.



**Fig. 2.** Imagen de luz estructurada infrarroja captada por Kinect.

En la figura anterior podemos ver una captura del patrón de luz estructurada emitido por Kinect. Se observa que en la mano la distancia entre los puntos es mayor que en el sofá posterior donde los puntos están más cerca. Estos son los datos a partir de los que se puede detectar e interpolar la distancia a todos los elementos de la escena.

## 4 ¿Qué es Intrael?

Es un programita pequeño que junto con los drivers de Kinect simplifica considerablemente su utilización, permite un uso muy sencillo, así como un uso distribuido.

Básicamente es un servidor web que utiliza http y JSON para interactuar con Kinect, realiza el procesamiento de la imagen, trabajando en torno al concepto de BLOBs. Intrael es un servidor que proporciona una interfaz HTTP para la Kinect. Procesa la información de profundidad que proporciona el dispositivo y lo segmenta a partir de un rango de profundidades o un fondo de referencia que no cambia. Posteriormente mide varias propiedades de los objetos aislados que detecta que llama blobs y los envía al cliente envueltos en un vector con formato JSON. Esta información puede ser cogida con XHR o recibida como video streaming a través de Server Sent Events.

Es posible realizar tareas de visión por ordenador desde un navegador con comandos AJAX plano. Esto abre una multitud de posibilidades para interacción en el desarrollo de aplicaciones web. El enfoque cliente servidor proporciona una gran versatilidad.

Los datos que se proporcionan de cada blob consisten en los puntos 3D de los extremos de x,y,z de los eje coordenados X e Y, junto la información geométrica del centro del objeto, así como con lecturas de profundidad de los correspondientes puntos del fondo. Finalmente se proporcionan conteo de los pixels que contiene el blob, para saber su tamaño aproximado por ejemplo, e información más general.

La información directa de la cámara también esta accesible en formato de imágenes JPEG, video MJPEG o binario directo. Esta última forma puede usarse con los vectores JavaScript u otro tipo proporcionado por el lenguaje con el que interactuemos con Intrael y nos permite realizar un procesado posterior y personalizado de la imagen en caso de ser necesario.

El servidor funciona perfectamente incluso en plataformas de pequeña potencia como la Beagleboard o la Raspberry Pi. Utiliza la libfreenect para la kinect, esto permite construir sistemas de muy bajo costo basado en Kinect. El código tiene la licencia de software libre AGPLv3.

#### 4.1 Instalación y uso

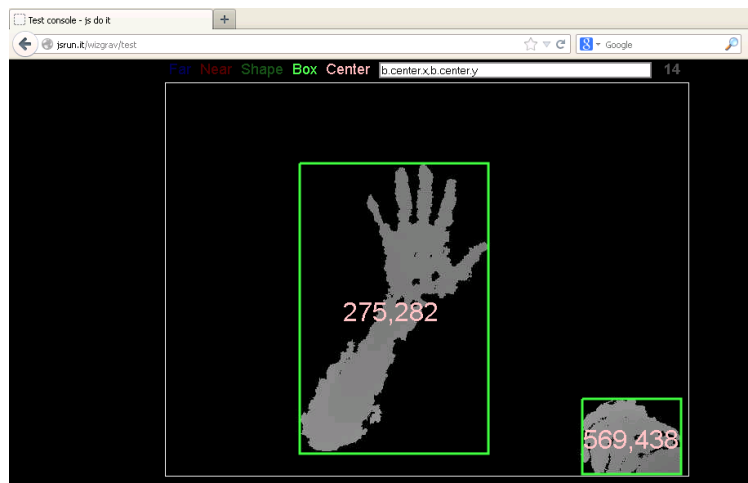
El proceso de instalación de Intrael resulta sumamente sencillo. Consiste básicamente en dos pasos: instalación de los drivers de Kinect y ejecución del programa `intrael.exe`. Para ello necesitamos bajar el archivo `.zip` con ambas cosas de la web de Intrael, cuya url es: <http://www.intrael.com>. Este archivo tendrá el nombre `intrael-1.0.wx86.zip`, no es necesario instalar, simplemente descomprimirlo en una carpeta de usuario cualquiera.

Una vez descomprimido los archivos, conectamos el dispositivo Kinect a nuestro equipo y nos solicitará los drivers que se encuentran en la carpeta `drivers` dentro de los archivos descomprimidos anteriormente. Necesitaremos instalar drivers para: cámaras, sonido y motor. Una vez instalados los drivers, simplemente hay que ejecutar el archivo `intrael.exe`. Este archivo viene con varias opciones de ejecución para cambiar, la IP, los puertos de escucha, claves, etc. Aunque en nuestro caso solo utilizaremos la opción `-s` que nos permitirá gestionar el programa a través de peticiones `http`. De tal forma que podemos comprobar el correcto funcionamiento de todo el proceso directamente con un navegador web, aunque es mucho más cómodo utilizar la aplicación de test que está accesible desde el sitio web de Intrael y que comentamos en el apartado siguiente.

#### 4.2 Aplicación de test

Para comprobar y monitorizar de forma cómoda y rápida el funcionamiento de Kinect con Intrael, han desarrollado una simple aplicación en javascript que realiza peticiones `http` y muestra la imagen con información sobre los blobs desde cualquier navegador. Dicha aplicación se encuentra en la siguiente dirección: <http://test.intrael.com>.

En la siguiente figura se muestra la aplicación de test de Intrael ejecutándose en un navegador web. Tenemos la opción de poder indicar en una línea de texto parámetros a ser visualizados junto con cada blob, así en la figura podemos ver que se ha introducido el texto: `b.center.x,b.center.y`. Donde `b` hace referencia al blob actual, `center` hace referencia al centro del mismo, `x` e `y` hacen referencia a las coordenadas del mismo. En nuestro caso vemos dos blobs uno que parece una mano abierta cuyas coordenadas del centro en la pantalla son 275 y 282, junto con otro blob que parece una mano cerrada cuyas coordenadas en pantalla son 569 y 438.



**Fig. 3.** Aplicación de ejemplo de uso de Intrael desarrollada en JavaScript y ejecutándose en un navegador.

Los blobs de la figura vienen rodeados por un rectángulo, pero también pueden ser rodeados por una figura formada por las coordenadas: máxima superior, inferior, derecha e izquierda.

## 5 Facilidad de uso de Kinect con Intrael

En este apartado vamos a mostrar lo sencillo que resulta interactuar con el entorno a través de Kinect utilizando la flexibilidad y versatilidad que nos ofrece el software Intrael.

### 5.1 Peticiones HTTP directas

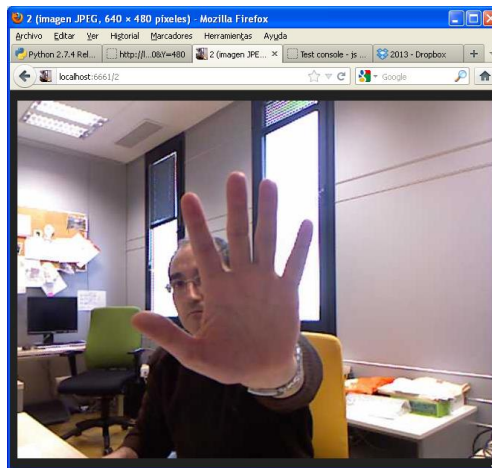
Desde cualquier navegador u aplicación que nos permita realizar peticiones http podemos gestionar y utilizar Intrael. La dirección a la que debemos enviar dichas peticiones será: `http://ip:6661/0`, donde IP representa la IP de la máquina sobre la que se está ejecutando Intrael, normalmente suele ser `localhost` si se está ejecutando el programa en el mismo ordenador desde el que se realizan las peticiones. El puerto donde está escuchando Intrael por defecto es `6661` y le debemos solicitar la carpeta `/0`, junto con una línea de órdenes codificadas con el protocolo CGI. Intrael

responderá con un array en formato texto JSON, con información general de Kinect junto con información referente a los blobs que ha detectado.

Como ejemplo, consideremos la siguiente petición realizada desde un navegador: `http://localhost:6661/0?a=15&z=600&Z=640`, le estamos solicitando a Intrael con  $a=15$  que gire  $15^\circ$  la cámara hacia arriba, con  $z=600$  que solo procese información de profundidad con valores superiores a 600 y con  $Z=640$  que sólo procese información de profundidad con valores inferiores a 640, de esta forma cualquier objeto fuera del rango 600-640mm será ignorado, mientras que los objetos dentro de ese rango serán procesados como blobs.

## 5.2 Ver imágenes

Intrael también nos permite acceder a las imágenes enviadas por Kinect, tanto la normal a color como la formada por la profundidad de la escena. Para ello simplemente hay que solicitar la carpeta /1 para la imagen a color y /2 para la imagen de profundidad. En la siguiente figura se muestra una captura de imagen a color desde un navegador simplemente accediendo a la url: `http://localhost:6661/1`.



**Fig. 4.** Ejemplo de una petición a Intrael desde un navegador solicitando una captura de la imagen de Kinect.

### 5.3 Aplicación demo

En este apartado vamos a mostrar lo fácil que resultaría crear una aplicación que interaccione con Kinect a través de Intrael, para ello utilizaremos el lenguaje Python, se puede utilizar cualquier otro lenguaje que nos permita realizar peticiones http.

La aplicación simplemente reconocerá un objeto, por ejemplo la palma de la mano, a través de un blob, en la escena y moverá una esfera en pantalla siguiendo las trayectorias.

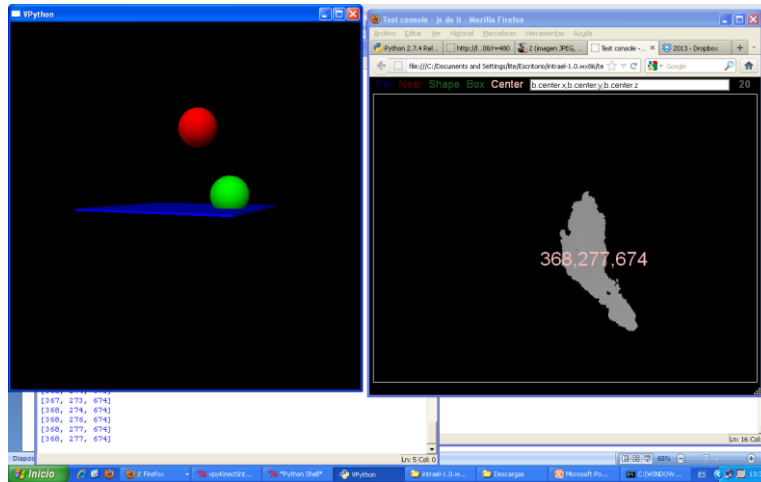
El siguiente código muestra la parte de la aplicación que interacciona con Intrael:

```
s=urllib2.urlopen("http://localhost:6661/0").read()
l=eval(s)
print l[16:19]
```

Como podemos ver, solamente son necesarias tres líneas: una primera donde se realiza la petición http, cuyo texto devuelto se guarda en la variable `s`; una segunda línea donde se procesa la cadena `s` y se convierte en un vector `l`, de números y una tercera donde se accede a las coordenadas `x`, `y`, `z` del primer blob, que corresponden con las posiciones 16,17 y 18 del array.

En la siguiente figura se muestran dos ventanas, una que muestra nuestra aplicación en funcionamiento y otra que se muestra la aplicación de test donde monitorizamos la información que está procesando Intrael. Vemos que la mano se encuentra en las coordenadas 268, 277 y 674 y la esfera roja, en este caso, también está posicionada en esas coordenadas. Si movemos nuestra mano, la bola cambia su posición acorde a la trayectoria de la misma.





**Fig. 5.** A la izquierda vemos la ventana correspondiente a nuestra aplicación de demostración y a la izquierda la aplicación de test que nos sirve para monitorizar el funcionamiento de Kinect con Intrael.

## 6 Proyectos con Intrael

Actualmente nos encontramos trabajando en varios proyectos que utilizan esta tecnología, como son: pyKinectIntrael, un wrapper para Intrael en Python, Evaluación de Intrael para interacción en Windows, Plataforma inteligente robotizada de procesos de laboratorio.

### 6.1 pyKinectIntrael

Estamos desarrollando una pequeña librería que permite interactuar con Intrael de manera más sencilla aun si cabe. Encapsula las peticiones http y nombre los campos de los blobs, nos indicara el color principal del blob. Permitirá interactuar con diversos blobs y diferenciarlos en función de su color. Se puede acceder al proyecto en la siguiente url: <http://launchpad.net/pykinectintrael>.

## 7 Conclusiones

En este trabajo se expone una tecnología novedosa denominada Intrael, que facilita la interacción con Kinect, permitiendo realizar una amplia gama de aplicaciones de forma muy sencilla. Si bien no tenemos acceso directo y control total a Kinect, sí que nos aporta una amplia funcionalidad genérica que permite interaccionar y monitorizar varios elementos independientes dentro de una escena, utilizando como elemento principal el blob.

## Referencias

- [1] Kinect for Windows. SDK Microsoft. Kinect // Microsoft.  
<http://www.microsoft.com/en-us/kinectforwindows/develop>
- [2] Pictures Kinect Visual Studio Developer Center.  
<http://msd.microsoft.com/en-us/vstudio/ff431702>.
- [3] Cómo funciona Kinect. Arbelaez Juan Pablo. 2011.  
<http://slideshare.net/AlbelaezGroup/kinect-commo-funciona-7228721>.
- [4] Intrael computer vision for the web. yannis Gravezas.  
<https://gcode.google.com/p/intrael>.

## **GreedExCol: un sistema interactivo colaborativo para el aprendizaje de algoritmos voraces**

Maximiliano Paredes-Velasco, Ouafae Debdi, J. Ángel Velázquez-Iturbide

Depto. Lenguajes y Sistemas Informáticos I

Universidad Rey Juan Carlos

Móstoles, Madrid, Spain

{maximiliano.paredes, ouafae.debdi, angel.velazquez}@urjc.es

**Abstract.** La motivación que tienen los alumnos en las tareas que deben abordar dentro del marco del proceso de aprendizaje es de vital importancia. Este es aun más importante cuando estamos en escenarios de aprendizaje activo, donde el rol del alumno es activo y el del profesor es de facilitador del proceso. Nosotros creemos que el estudio y exploración de la motivación del alumno en el proceso de aprendizaje puede aportar ideas y nuevos caminos que ayuden a mejorar la efectividad de aprendizaje, especialmente en dominios CSCL. En este artículo presentamos una herramienta colaborativa para el aprendizaje de la algoritmia, GreedExCol y una evaluación de la misma desde el enfoque de la motivación de los alumnos.

**Keywords:** CSCL, Motivación, Algoritmia.

### **1 Introducción**

La motivación que tienen los alumnos en el proceso de aprendizaje se debe tener en cuenta ya que ayuda a obtener resultados positivos en el proceso de aprendizaje [1] produciendo un aprendizaje de alta calidad y con creatividad [2]. Centrándonos en los modelos instruccionales de aprendizaje activo, la motivación constituye un elemento nuclear que mejora el proceso de aprendizaje-enseñanza [3]. Estos modelos instruccionales se aplican ampliamente en CSCL, cobrando por tanto importancia la motivación en este marco. Pese a la importancia que tiene la motivación en el proceso aprendizaje-enseñanza en CSCL no hay trabajos ni experiencias en la literatura actual bajo un enfoque empírico que estudien la utilización de las herramientas y sus

implicaciones en la motivación del alumno al usar esas herramientas. Esta carencia se acentúa más aun cuando nos centramos en el dominio del aprendizaje colaborativo de la programación y la algoritmia. Las experiencias realizadas con las herramientas actuales para el aprendizaje de las materias de programación se centran en otros aspectos diferentes de la motivación (eficiencia de aprendizaje, usabilidad, etc.), dejando éste a un lado. Nosotros pensamos que la realización y estudio de experiencias CSCL en las que se analicen la relación de las herramientas que utiliza el alumno y la motivación que experimenta éste al utilizarlas puede ser de gran interés para la comunidad de investigadores. En este trabajo se presenta el estudio de motivación que se ha realizado en el aprendizaje de algoritmos voraces utilizando la herramienta GreedExCol, un sistema interactivo colaborativo para el aprendizaje de algoritmos voraces mediante el método experimental.

En la sección 2 del artículo revisamos las principales herramientas de aprendizaje de programación y las experiencias realizadas en el aula en las que se evalúan la motivación de los alumnos. En la sección 3 describimos el sistema GreedExCol para posteriormente en la sección 4 mostrar la experiencia realizada con dicho sistema. El artículo termina con una sección de conclusiones.

## **2 Revisión de trabajos relacionados**

En esta sección presentamos las herramientas más representativas para el aprendizaje de la programación en general así como las experiencias realizadas en el aula en las que se ha evaluado la motivación de los alumnos en el proceso de aprendizaje en diferentes dominios y materias.

### **2.1 Herramientas para el aprendizaje de programación**

Existen numerosos tutores y herramientas para el aprendizaje/enseñanza de la programación basadas en interacción individual del alumno con la herramienta. Sin embargo hay muy pocas cuando nos centramos en una interacción colaborativa en la que varios alumnos tienen que utilizar a la vez la herramienta. De acuerdo a la taxonomía de sistemas CSCW de [4] presentamos una revisión de herramientas centrándonos en las herramientas de tipo Interacción Distribuida Asíncrona e Interacción Distribuida Síncrona. Dentro de la taxonomía de Johansen, GreedExCol se

puede clasificar principalmente como un sistema de Interacción Cara a Cara, sin embargo presenta funcionalidades de interacción distribuida síncrona y asíncrona, por lo que creemos que es conveniente centrar el estudio en este tipo de herramientas. Algunas herramientas de aprendizaje colaborativo presentan características de entornos IDE (Integrated Development Environment), soportando funcionalidades propias de entornos de desarrollo profesionales. Por ejemplo, SICODE [5] es una herramienta Web de desarrollo de software para aprendizaje del lenguaje Java o COLLEGE [6] que soporta compilación y ejecución. Estos sistemas, si bien presentan utilidades de toma de decisión para funciones concretas como son compilar o ejecutar, no tienen utilidades para la toma de decisión sobre el diseño del programa en general.

Algunas herramientas no sólo están diseñadas a partir de IDEs profesionales si no que están integrados en los propios IDEs. Éste es el caso por ejemplo de las herramientas que están desarrolladas como Plug-in de Eclipse. Por ejemplo, COALA [7] es un sistema distribuido que permite que el profesor haga anotaciones a mano alzada sobre el código fuente que desarrolla el alumno, o la herramienta Cole-Programming [7], que extiende el plug-in de COALA con utilidades de interacción colaborativa proporcionando un Chat y un Foro. La herramienta Jazz Samgan [8] es un ejemplo más de este tipo de herramientas. Estas herramientas en general se usan como repositorios para que los alumnos se conecten y descarguen un proyecto (con una plantilla explícita del correspondiente plug-in) y a partir de él hagan la tarea de codificación pura y dura sin guías que los dirijan. Esto conlleva que en muchos casos estas herramientas no soporten funcionalidades como la gestión de grupos y enunciados y la edición colaborativa de una misma fuente por todos los miembros del grupo. Incluso en algunas herramientas el trabajo colaborativo se reduce a la realización de una misma tarea por todo los miembros del grupo de forma independiente y una vez finalizada cada uno de los alumnos del grupo envía su código fuente resultante al profesor más que aprendizaje colaborativo podríamos hablar de aprendizaje cooperativo. Este es el caso de por ejemplo COALA y Cole-Programming.

Otras herramientas soportan capacidades de visualización y animación del código fuente para que los alumnos puedan entender mejor el comportamiento de los algoritmos. Éste es el caso de por ejemplo JeCo [9] y SICAS-COL [10]. En algunos casos estas funcionalidades de visualización y simulación se consiguen como resultado de la fusión de otras herramientas. Por ejemplo, la herramienta SICAS-COL es el resultado de fusionar dos herramientas: PlanEdit [11] y SICAS [11], y la herramienta JeCo es la fusión de Jeliot 3 [12] y Woven Stories [13]. Este tipo de herramientas suelen

visualizar gráficamente el diseño del programa (mediante diagramas de flujo como por ejemplo hace SICAS-COL) y visualizar el código fuente y hacer animación del mismo visualizando gráficamente el comportamiento de las expresiones, estructuras de datos y la pila de ejecución.

Como ya se ha dicho anteriormente estas herramientas son la fusión de, por un lado herramientas para la colaboración de dominios no propios de programación (como por ejemplo Woven Stories, que es de dominio de escritura, o PlanEdit, que fue diseñada para el dominio de la domótica, y por el otro lado herramientas que han sido diseñadas para la animación y visualización pero con un uso individual, como Jeliot 3 o SICAS.

Estas fusiones presentan algunas limitaciones en su integración, como por ejemplo que las utilidades de discusión y argumentación como por ejemplo el chat no están enlazadas con el código desarrollado.

Otras herramientas centran el proceso de aprendizaje en la adquisición de conceptos básicos como pueden ser ámbito y visibilidad de variables como MoCAS [14] o adquisición de habilidades relacionadas con la programación como HabiProt [15]. Estas herramientas están más orientadas a ser repositorios de acceso colaborativo y no relacionan ni estructuran debidamente las aportaciones de los fragmentos de código.

Las herramientas que hemos visto son desarrolladas en su mayoría para plataforma PC, bien sea ad-hoc o bien con tecnología Web. Otras herramientas están orientadas a plataformas móviles, como MSA y MoCAS o a plataformas wiki como Adessowiki [16]. Estas herramientas no contemplan funcionalidades típicas para el aprendizaje de la programación como son la depuración de programa.

## **2.2 Experiencias de estudio de la motivación en CSCL**

Los principales instrumentos de evaluación que se utilizan en las experiencias CSCL son pretest y postest para evaluar la eficacia educativa, SNA (Social Network Analysis) para medir la dinámica de los procesos de aprendizaje, técnicas de Análisis de Contenidos para analizar el discurso en CSCL y con menor presencia escalas como AMS (Academic Motivation Scale) y OCLEFQ (Online Collaborative Learning Experiences Frustration Questionnaire) para medir aspectos emotivos. A nuestro entender no hay estudios que analicen las implicaciones y relaciones de la motivación con la usabilidad de herramientas CSCL en el dominio de aprendizaje de la programación. Haciendo una revisión de la cuestión de experiencias CSCL más abierta,

sin centrarnos en el dominio de la programación, podemos encontrar escasas experiencias que han estudiado la implicación de la motivación. Algunas experiencias realizadas se han centrado en estudiar la implicación o relación que hay entre la motivación y la interacción entre los miembros del grupo (Ref. e1, e3, e2) [17] [19][18]. Estos trabajos analizan el diálogo entre los miembros bajo dos posibles enfoques. El primer enfoque se centra en identificar roles dentro del grupo a partir del diálogo e interacción que se produce entre los miembros (ej. e1) [17]. Para ello se establecen unos roles iniciales a propuesta del profesor, como son el rol de coordinador del grupo, rol de miembro no participativo, etc., y mediante las intervenciones de los alumnos se van asociando a los alumnos diferentes roles. En algunos casos pueden aparecer nuevos roles que no hayan sido definidos inicialmente. A partir de estos roles se intenta estudiar la relación de la motivación y las aportaciones en el discurso. El segundo enfoque se centra en categorizar las intervenciones de los alumnos en el diálogo. Las intervenciones se clasifican en al menos en 1) relacionadas con el dominio (la tarea que tienen que realizar) y 2) las no relacionadas (en algunos casos se establece una clasificación de hasta 9 categorías como es el caso de e2) [18]. A partir de estas categorías se analiza la influencia de la motivación en las contribuciones del diálogo y la discusión. Estas experiencias se basan en utilizar instrumentos de Análisis de Contenidos y SNA (Social Network Analysis) para analizar la interacción que se produce en el grupo, ya sea a partir de roles o de categorías, generando grafos que modelan el discurso y permiten ver visualmente el cuerpo de la discusión.

Otros trabajos se han centrado en estudiar la relación que tienen las técnicas de scaffolding y la motivación; éste es el caso de (e2) [18]. Esta experiencia analiza la motivación desde el punto de vista en que la motivación está relacionada con el grado de libertad que tiene el alumno para realizar tareas, y éste está regulado a su vez por el nivel de scaffolding que apliquemos. La motivación es analizada en varias dimensiones (tanto la motivación intrínseca como extrínseca) y como instrumento usan la (Academic Motivation Scale), utilizando un grupo de control (totalmente autónomos) y grupos experimentales (dirigidos por scaffolding). Las experiencias realizadas son en el dominio del aprendizaje de la economía.

Hay trabajos que exploran la relación del entorno en que se encuentra el alumno con la motivación (e4, e5) [20,21]. Estos trabajos han realizado experiencias en el dominio del aprendizaje de las matemáticas. En algunos casos se han desarrollado experiencias que comprenden diferentes escenarios de aprendizaje como son en la oficina, en el colegio y en la universidad y se han comparado la interacción resultante. Como

instrumento de medida se utiliza la opinión de los alumnos sobre la colaboración, en algunos casos mediante cuestionarios en los que tienen que valorar en un rango aspectos como son el grado en que tu compañero ha colaborado, ha trabajado, etc. Estos trabajos no son muy determinantes y las conclusiones que arrojaron fueron que la motivación del estudiante puede afectar a la percepción de lo que hacen los compañeros en el proceso de colaboración y a la percepción del entorno colaborativo (e4) [20].

Otro trabajo que se ha realizado recientemente en el dominio que nos ocupa es e6) [22]. Este trabajo ha desarrollado una experiencia con alumnos de un master universitario en la tarea de escritura colaborativa en la que se ha estudiado el grado de frustración que experimentan los alumnos en las tareas de colaboración. En este caso no se utilizan escalas de motivación (tipo AMS) si no que los autores han desarrollado una escala específica de frustración (Online Collaborative Learning Experiences Frustration Questionnaire – OCLEFQ). Aspectos como la colaboración asimétrica entre los miembros del grupo, las barreras que ponen los sistemas Online colaborativos en la comunicación entre los alumnos o el desequilibrio en el nivel de compromiso son aspectos que hacen que los alumnos experimenten frustración.

En conclusión, podemos ver que el análisis de la motivación en las experiencias se ha realizado diferenciando la motivación intrínseca y la extrínseca (e4, e7, e2) [20, 23, 18], incluso en algunos casos se ha analizado varias dimensiones de la motivación (e7, e2) [23, 18]. Sin embargo hay otras experiencias que no hacen un análisis de grano tan fino (por ejemplo e5 o e3) [21, 19].

Desde el punto de vista de la intervención de la tecnología en el aprendizaje para la evaluación de la motivación hemos de señalar que hay experiencias con sistemas online puro en el que no hay contacto visual entre los alumnos durante todo el curso (e2, e7) [18, 23] y en otros la interacción es cara a cara mediando tecnología como un soporte más de ayuda para las tareas (e1, e3, e4, e5, e6) [17, 19,-22].

Los instrumentos de medición que se utilizan para la motivación en estos escenarios educativos son entrevistas o encuestas de opinión (e5) [21] o escalas de motivación como la Academic Motivation Scale (e2, e4, e7) [18, 20, 23]. En algunos casos se utilizan cuestionarios específicos de colaboración (e4) [20] o de frustración (e6) [47]. Si bien la práctica más interesante es la utilización de técnicas multi-método que combinan diferentes instrumentos como AMS (Academic Motivation Scale), SNA (Social Network Analysis) y CA (Content Analysis) (e7 y e2) [23, 18]. De forma minoritaria, en algunos casos estos instrumentos se combinan con pretest y posttest de conocimiento (e4) [20].



Desde el punto de vista de materias de aprendizaje en las que se ha evaluado la motivación hemos de señalar que han sido muy escasas, centrándose en ciencias (e2, e4, e7) [18, 20, 23] (economía y matemáticas) y en letras (e3, e6) [19, 22] (historia y escritura de documentos). En el dominio del aprendizaje de la programación no hay ninguna experiencia en la revisión que hemos realizado nosotros y sólo hemos encontrado una en dominio del aprendizaje de la tecnologías (e1) [17] de creación de contenidos Web.

Por tanto se desprende de esta revisión que hemos presentado, que no hay experiencias que hayan abordado el estudio de la motivación del alumno en el campo del aprendizaje de la programación ni de la algoritmia.

### 3 Sistema interactivo colaborativo GreedExCol

GreedExCol es un sistema interactivo para el aprendizaje de algoritmos voraces que soporta un espacio de contribuciones que facilita la discusión y toma de decisiones face-to-face dentro de un grupo de alumnos sobre la solución de un problema. Cada alumno trabaja de forma individual en las primeras etapas de experimentación y posteriormente interviene en un debate con el resto de los miembros del grupo para decidir qué funciones de selección son óptimas a partir de los datos obtenidos por todos ellos. Para realizar este debate, los miembros de un grupo comparten los datos obtenidos experimentalmente, mostrados en las visualizaciones y en las tablas.

El alumno comienza a experimentar con diferentes estrategias de selección para intentar identificar la óptima si la hay. Una vez que ha realizado los experimentos puede entrar en el espacio grupal al seleccionar la opción de menú “Discusión” y regresar a su espacio individual pulsando la opción de menú “Cerrar vista discusión” (Figura 1). También puede regresar sin salir del espacio de propuestas, seleccionando la pestaña que lleva su nombre en la barra de botones (véase Figura 1, etiqueta C).

En la Figura 1 vemos el espacio colaborativo junto con una nueva barra de botones que muestra:

1. El botón “Comparativa” permite mostrar simultáneamente los espacios individuales de trabajo de cada uno de los miembros del grupo (véase Figura 1, etiqueta A). La vista que muestra como se puede ver en la Figura 1 es la visualización gráfica y de tabla de los experimentos de cada miembro.

2. Hay tantos botones como número de miembros hay en el grupo. Al seleccionar uno de estos botones, se muestra el espacio personal de trabajo del miembro del grupo seleccionado (Figura 1, etiqueta C).
3. El botón “Actualizar el trabajo de los compañeros” permite que bajo demanda se actualicen los datos de todo el grupo, es decir, sus últimas ejecuciones, tablas y propuestas, actualizándose la vista de la pantalla con los nuevos experimentos realizados (véase Figura 1, etiqueta D).
4. Propuesta. GreedExCol permite a los alumnos utilizar el envío de propuestas como método de comunicación entre los miembros del grupo. Una propuesta es un experimento formado por los datos de una o varias ejecuciones de un compañero. Cada propuesta se acompaña de un mensaje para que el receptor de la propuesta pueda saber qué quiere decir su compañero con el experimento enviado. Al seleccionar la propuesta de un compañero del grupo (véase Figura 1, etiqueta B), la pantalla muestra el espacio de ejecución del experimento enviado (la vista gráfica y las tablas). En esta nueva vista el receptor podrá ver el mensaje (de texto) que acompaña a la propuesta, aceptarla y almacenarla permanentemente en el espacio individual del alumno receptor o rechazarla y eliminar definitivamente la propuesta.

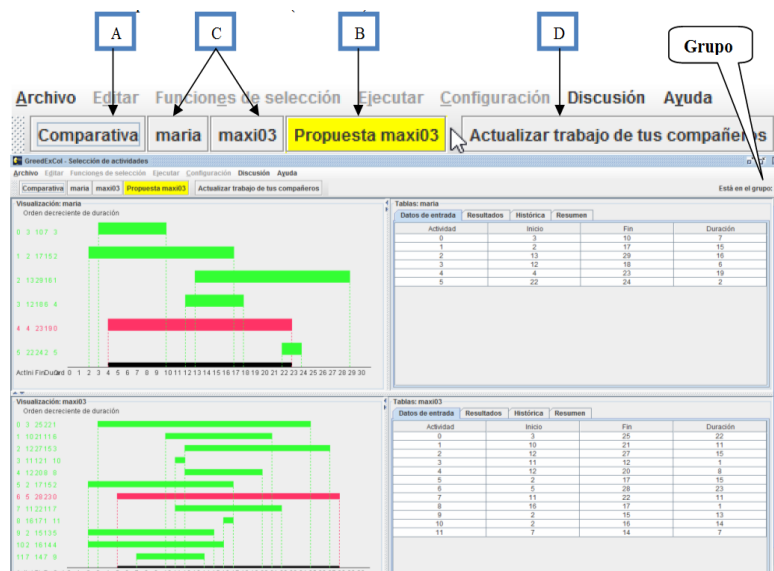


Fig. 1. Espacio colaborativo de GreedExCol.

## **4 Experiencia en el aula de aprendizaje de algoritmos voraces**

Hemos llevado a cabo una experiencia de uso con GreedExCol en la que hemos evaluado el grado de usabilidad y de motivación que tiene la herramienta en los alumnos.

### **4.1 Descripción de la experiencia**

El objetivo principal de la realización de la experiencia es medir la usabilidad y la motivación que tienen los estudiantes utilizando la herramienta GreedExCol. La asignatura utilizada para realizar la experimentación fue Diseño y Análisis de Algoritmos del Grado de Ingeniería del Software de la Universidad Rey Juan Carlos. La motivación que experimentan los alumnos se midió con el instrumento de una escala de motivación situacional llamada EMSI. La usabilidad que perciben los alumnos de la herramienta GreedExCol fue medida con un cuestionario de usabilidad derivado de SUS (System Usability Scale). Esta usabilidad fue medida en la utilización de la GreedExCol. El profesor usó la herramienta a modo de ejemplo en dos problemas: 1) el problema de maximizar el número de objetos en la mochila y 2) maximizar el peso en la mochila. En estas sesiones el profesor explicó algunos conceptos básicos de la experimentación en algoritmos y los alumnos se familiarizaron con el método experimental y la utilización de GreedExCol. La evaluación se realizó en dos sesiones de laboratorio. En la primera los estudiantes experimentaron con la herramienta GreedExCol en el problema de la mochila, teniendo como objetivo identificar la función de selección óptima. En la segunda sesión los alumnos tuvieron que realizar la misma tarea pero para el problema de selección de actividades. Al final de la segunda sesión los estudiantes entregaron un informe describiendo sus experimentos y la función de selección óptima que proponían así como al realización de los cuestionarios de evaluación de usabilidad y de motivación.

### **4.2 Resultados**

Como ya se ha comentado anteriormente el tamaño de la muestra es 27 miembros. En el estudio y medición de la motivación se identificaron cinco variables dependientes: una por cada dimensión de la motivación de acuerdo a la teoría self-determination

(intrínseca, regulación identificada, regulación externa y desmotivación) más una quinta que se corresponde con la motivación en global como resultado de calcular las medias de las cuatro dimensiones anteriores (denominada Media Motivación). Las variables independientes del experimento fueron 14, que se corresponde con los 14 ítems del cuestionario que se utilizó. Estas variables se midieron en una escala Link de 1 a 7, siendo el valor 1 la valoración más negativa y 7 la más positiva. El instrumento utilizado fue la escala EMSI. En este instrumento se proporciona al participante una pregunta sobre la actividad que ha realizado y 14 posibles afirmaciones como respuestas a esa pregunta. El alumno debe valorar el grado en el que está de acuerdo con la propuesta, siendo 1 el valor que indica que no está de acuerdo con la afirmación, 3-4 indican que está más o menos 50% de acuerdo y 7 que está totalmente de acuerdo. La pregunta que se les realizó para medir la motivación a los alumnos fue:

*¿Por qué crees que deberías realizar la actividad de usar GreedEx para estudiar los algoritmos voraces?*

En el estudio y medición de la interfaz de usuario se utilizó un cuestionario compuesto de 19 ítems con valores en una escala de Likert de 1 (muy mala) a 5 (muy buena) para valorar la usabilidad de la interfaz. Estos ítems están orientados a valorar aspectos como el grado de complejidad, facilidad, integración de las funcionalidades, etc. de la herramienta. Por tanto, en el estudio de la interfaz identificamos 19 variables independientes (una por cada ítem). El cuestionario está inspirado en la escala System Usability Scale (SUS). Hemos decidido realizar un cuestionario nuevo inspirándonos en la escala SUS y no utilizar la propia escala porque consideramos necesario ítems que midan aspectos concretos sobre algoritmos voraces, como por ejemplo el grado en que la herramienta ayuda al alumno a identificar la estrategia óptima, a analizar el efecto de las distintas estrategias voraces, etc. Las variables medidas en el estudio de la usabilidad se agrupan en tres categorías con el objetivo de facilitar su presentación (entre paréntesis se indica el nombre de la variable independiente):

1. Aspectos visuales de la interfaz. Se mide la calidad entendida como la facilidad de uso y como la capacidad de integración de las funcionalidades en la interfaz de los siguientes componentes de la interfaz:
  - Los iconos (Iconos)
  - Panel de visualización del gráfico (PanelVisualización)

- Pestaña que muestra el enunciado del problema (P. Problema)
  - Pestaña que muestra el algoritmo voraz (P. Algoritmo)
  - Estructura del menú (Estructura)
  - Pestaña que muestra la tabla de datos de entrada (T. Entrada)
  - Pestaña que muestra la tabla histórica de las ejecuciones de las diferentes estrategias con los % de optimalidad (T. Abreviada)
  - Pestaña que muestra la tabla de resultados con las soluciones de una ejecución para cada estrategia (T. Resultados)
  - Tabla de resumen que muestra para cada estrategia el valor de la función objetivo para todas las ejecuciones (T. Resumen)
2. Aspectos funcionales:
- Facilidad en usar la herramienta (Facilidad)
  - Facilidad y buena funcionalidad para la exportación de imágenes y tablas (Exportación)
  - Facilidad para poder seleccionar diferentes estrategias voraces (FuncionesSelección)
  - Utilidad para analizar el efecto de las estrategias (Analizar)
  - Utilidad para identificar la estrategia óptima (Identificar)
  - Facilidad y buena funcionalidad para analizar las estrategias voraces (Calidad)
  - Satisfacción de GreedExCol en general (TGustaGreedExCol)
3. Otros aspectos:
- Facilidad y funcionalidad para introducir y generar datos de entrada (Introducción/GeneraciónDatos)
  - Facilidad y buena funcionalidad para realizar ejecuciones del algoritmo y animaciones (Ejecución/Animación)
  - Calidad de los servicios de discusión en grupo (MenúDiscusión)

El análisis estadístico inicial que se hizo fue en primer lugar comprobar la normalidad de la muestra para las variables independientes de la usabilidad, las variables dependientes de las dimensiones de motivación y la variable dependiente de la media de la motivación. Ésta última variable se calcula como la media aritmética de las 12 variables independientes de motivación teniendo en cuenta que se ha invertido el

valor de las variables correspondiente a los ítems negativos (3, 4, 7, 8, 10 y 14). Para saber si la muestra es normal se aplicó el test de Kolmogorov-Smirnov con un error estándar del 5% ( $p=0.05$ ). En este test se establece como hipótesis nula *Ho: la muestra es normal*. La tabla 1 muestra el resultado de la normalidad para las variables medidas en el estudio de la usabilidad (Tabla 1, desde la fila 2 a la 20) y la variable de la media de motivación (Tabla 1, fila 1). Podemos ver que  $sig>0.05$  (celdas en negrita de la Tabla 1) por lo que no se rechaza la hipótesis nula, por lo que asumimos que la muestra es normal para las variables de usabilidad y de la media de motivación.

**Tabla 1.** Análisis de la normalidad de las variables de usabilidad y variable de media de la motivación.

	N	Parámetros normales <sup>a,b</sup>		Diferencias más extremas			Z de Kolmogorov-Smirnov	Sig. asintótico (bilateral)
		Media	Desviación típica	Absoluta	Positiva	Negativa		
<b>MediaMotivación</b>	27	4,0582	,77288	,138	,087	-,138	,718	<b>,682</b>
<b>Facilidad</b>	27	5,9111	1,30718	,279	,202	-,279	1,450	<b>,060</b>
<b>Analizar</b>	27	5,8074	1,14554	,243	,201	-,243	1,263	<b>,083</b>
<b>Identificar</b>	27	6,1704	1,11590	,364	,229	-,364	1,891	<b>,200</b>
<b>Calidad</b>	27	5,4444	1,18625	,219	,189	-,219	1,137	<b>,151</b>
<b>Estructura</b>	27	4,8222	1,56926	,210	,210	-,161	1,090	<b>,186</b>
<b>Iconos</b>	27	4,5630	1,62863	,182	,144	-,182	,947	<b>,331</b>
<b>PanelVisualización</b>	27	5,1333	1,28781	,234	,173	-,234	1,216	<b>,104</b>
<b>TablaResultados</b>	27	4,7704	1,51678	,226	,144	-,226	1,176	<b>,126</b>
<b>T. Resumen</b>	27	5,0815	1,29646	,322	,233	-,322	1,674	<b>,077</b>
<b>T. Abreviada</b>	27	5,1852	1,63888	,199	,171	-,199	1,035	<b>,234</b>
<b>Introducción/generación</b>	27	4,6667	1,50384	,177	,177	-,177	,922	<b>,363</b>
<b>Ejecución</b>	27	5,0815	1,35335	,205	,187	-,205	1,064	<b>,208</b>
<b>FuncionesSelección</b>	27	5,3407	1,34715	,209	,209	-,187	1,085	<b>,190</b>
<b>Exportación</b>	27	5,1852	1,59222	,195	,139	-,195	1,015	<b>,254</b>

<b>MenúDiscusión</b>	27	4,9778	1,41784	,227	,227	-,151	1,179	<b>,124</b>
<b>TGustaGreedExCol</b>	27	5,2370	,99698	,309	,247	-,309	1,604	<b>,072</b>
<b>P. Problema</b>	27	5,0296	1,41389	,203	,203	-,175	1,054	<b>,217</b>
<b>P. Algoritmo</b>	27	5,2889	1,41784	,260	,260	-,220	1,352	<b>,052</b>
<b>T. Entrada</b>	27	5,3926	1,38395	,226	,144	-,226	1,176	<b>,126</b>

Este mismo test se aplica para estudiar la normalidad de las cuatro variables dependientes de las dimensiones de la motivación. Cada una de estas variables se calcula como la media de las variables independientes correspondientes a la dimensión concreta. Por ejemplo, la variable Motivación Intrínseca se calcula con la media de los valores de los ítems 1, 5, 9 y 11 del cuestionario. Las dimensiones que contiene ítems negativos (la motivación regulada externa y la desmotivación) no se han invertido sus valores. La tabla 2 muestra el resultado de la normalidad. Podemos ver que  $\text{sig} > 0.05$  por lo que no se rechaza la hipótesis nula, por lo que asumimos que la muestra es normal.

**Tabla 2.** Prueba de normalidad para las cuatro dimensiones.

Prueba de Kolmogorov-Smirnov		Intrínseca	Identificada	Externa	Desmotivació n	
Número alumnos		27	27	27	27	
Parámetros normales <sup>a,b</sup>		Media	3,45370	4,01235	4,62963	3,19444
		Desviación típica	,890757	1,265862	1,166972	1,463224
Diferencias extremas	más	Absoluta	,080	,126	,147	,172
		Positiva	,080	,104	,088	,172
		Negativa	-,076	-,126	-,147	-,108
Z de Kolmogorov-Smirnov		,416	,653	,762	,891	
Sig. asintót. (bilateral)		<b>,995</b>	<b>,787</b>	<b>,607</b>	<b>,405</b>	

En este primer análisis de los resultados de motivación podemos ver que las muestras siguen una distribución normal por lo que podemos aplicar cálculos de correlación a estas variables.

## 5 Conclusiones

En este artículo hemos presentado la herramienta GreedExCol que explota las posibilidades de CSCL aplicadas al aprendizaje de la algoritmia, en concreto al esquema voraz. Este sistema proporciona una herramienta interactiva que permite al alumno aprender mientras realiza experimentos con diferentes funciones de selección de un algoritmo voraz. En este proceso de experimentación la motivación del sujeto es de vital importancia. Por ello hemos realizado una experiencia en el aula en la que hemos medido por un lado la usabilidad de GreedExCol y por otro el grado de motivación del alumno en participar y utilizar la herramienta para aprender algoritmos voraces.

Como conclusión hemos podido ver que las muestras obtenidas y los instrumentos utilizados para obtener dichas muestras son correctos. Hemos aplicado test estadísticos de normalidad y hemos concluido que son muestras normales. Se abre ahora un camino de análisis estadísticos en los que podemos aplicar métodos estadísticos sobre muestras normales que nos permitan concluir más elaboradamente resultados de la experiencia realidad. Esto se traduce en un trabajo futuro que esperamos abordar en breve.

## Agradecimientos

Este trabajo ha sido desarrollado por el proyecto de investigación TIN2011-29542-C02-01 del Ministerio de Educación, Cultura y Deporte de España.

## Referencias

- [1] Ryan, R. M., & Deci, E. L. (2000). Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being. *American Psychologist*, 55, 68-78.
- [2] Ryan, R. M., & Deci, E. L. (2000b). Intrinsic and extrinsic motivations: Classic definitions and new directions. *Contemporary Educational Psychology*, 25, 54-67
- [3] Pintrich, P. (2003). Motivation and classroom learning. In W. Reynolds & G. Miller (eds.), *Handbook of Psychology: Educational Psychology*, Vol. 7 (pp. 103-122). Hoboken, NJ: John Wiley & Sons.



- 
- [4] Johansen R., Sibbet D., Benson S., Martin A., Mittman R. y Saffo P.(1991). *Leading Business Teams*. Addison Wesley
- [5] Crescencio Bravo, Maria José Marcelino, Anabela Gomes, Micaela Esteves, António José Mendes.: *Integrating Educational Tools for Collaborative Computer Programming Learning*. *Journal of Universal Computer Science - J.UCS* , vol. 11, no. 9, pp. 1505-1517. (2005)
- [6] Aurora Vizcaíno, Juan Contreras, Jesús Favela, Manuel Prieto.: *An Adaptive, Collaborative Environment to Develop Good Habits in Programming*. *The 5th International Conference on Intelligent Tutoring Systems*. Pages 262-271. Springer-Verlag London, UK (2000)
- [7] Amrit, Chintan and Hillegersberg van, Jos and Harmsen, Frank.: *1st International Workshop on Tools for Managing Globally Distributed Software Development (TOMAG 2007)*. *The Seventh International Conference of Computer Ethics: Philosophical Enquiry*. CTIT workshop proceedings series. Center for Telematics and Information Technology, CTIT, Enschede (2007)
- [8] Niko MYLLER Jussi NUUTINEN.: *JeCo: Combining Program Visualization and Story Weaving*. *Informatics in Education*, Vol. 5, No. 2, 255–264 255 (2006)
- [9] P. Gerdt, P. Kommers, C. K. Looi, and E. Sutinen. *Woven stories as a cognitive tool*. In *Cognitive Technology*, LNAI 2117, pages 233–247, 2001.
- [10] Moreno, A., N. Myller, E. Sutinen, M. Ben-Ari.: *Visualizing programs with Jeliot 3*. In *Proceedings of the International Working Conference on Advanced Visual Interfaces AVI 2004*. Gallipoli (Lecce), Italy. pp. 373–376 (2004)
- [11] Santos, Á.; Gomes, A.; Mendes, A.J.: *Integrating New Technologies and Existing Tools to Promote Programming Learning*. *Algorithms*. 3, 183-196. (2010)
- [12] Moreno, A., N. Myller, E. Sutinen, M. Ben-Ari.: *Visualizing programs with Jeliot 3*. In *Proceedings of the International Working Conference on Advanced Visual Interfaces AVI 2004*. Gallipoli (Lecce), Italy. pp. 373–376 (2004)
- [13] P. Gerdt, P. Kommers, C. K. Looi, and E. Sutinen. *Woven stories as a cognitive tool*. In *Cognitive Technology*, LNAI 2117, pages 233–247, 2001.
- [14] Luis Miguel Serrano Cámara, Maximiliano Paredes Velasco, Carlos M<sup>a</sup> Alcover and J. Ángel Velázquez Iturbide, "An evaluation of students" motivation in computer-supported collaborative learning of programming concepts", *Computers in Human Behavior*, in press.
- [15] Aurora Vizcaíno, Juan Contreras, Jesús Favela, Manuel Prieto.: *An Adaptive, Collaborative Environment to Develop Good Habits in Programming*. *The 5th*

- International Conference on Intelligent Tutoring Systems. Pages 262-271. Springer-Verlag London, UK (2000)
- [16] Leticia Rittner, André Vital Saúde, Alexandre G. Silva, Rubens Campos Machado, Mariana P. Bento, Roberto de Alencar Lotufo. "Adessowiki: Collaborative Scientific Programming Environment", 24th SIBGRAPI Conference on Graphics, Patterns, and Images Tutorials (2011)
- [17] [e1] Marcos García, J.A., Martínez Monés, A., Dimitriadis, Y., Anguita Martínez, R., Ruiz Requies, I., Rubia Avi, B.: Detecting and Solving Negative Situations in Real CSCL Experiences with a Role-Based Interaction Analysis Approach in Intelligent Collaborative e-learning Systems and Applications. Serie Studies in Computational Intelligence, Vol. 246, Springer-Berlag, Berlin-Heidelberg (2009)
- [18] [e2] Rienties, B., Giesbers, B., Tempelaar, D. T., Lygo-Baker, S., Segers, M., & Gijsselaers, W. H.: The role of scaffolding and motivation in CSCL. *Computers & Education*, 59(3), 893-906. DOI: 10.1016/j.compedu.2012.04.010. (2012)
- [19] [e3] Tapola, A., Hakkarainen, K., Syri, J., Lipponen, L., 3Palonen, T. & \_Niemi-virta M.: Motivation and Participation in Inquiry Learning within a Networked Learning Environment. Centre for Research on Networked Learning and Knowledge Building Department of Psychology (University of Helsinki). <http://www.helsinki.fi/science/networkedlearning/texts/tapolaetal2001.pdf>
- [20] [e4] Howley, I., Chaudhuri, S., Kumar, R., & Rosé, C.: Motivation and Collaborative Behavior: An Exploratory Analysis. Poster paper presented at Computer Supported Collaborative Learning (2009)
- [21] [e5] Eales, R. T. J., Hall, T. and Bannon, L. J.: The Motivation is the Message: Comparing CSCL in different settings. In the proceedings of CSCL2002 (International conference on Computer Support for Collaborative Learning), Boulder, Colorado, USA (2002)
- [22] [e6] Capdeferro, N., & Romero, M.: Are online learners frustrated with collaborative learning experiences?. *The International Review Of Research In Open And Distance Learning*, 13(2), pp. 26-44 (2012)
- [23] [e7] Rienties, B, Tempelaar, DT, Van den Bossche, P, Gijsselaers, WH and Segers, M.: The role of academic motivation in Computer-Supported Collaborative Learning. *Computers in Human Behaviour*, 25 (6). 1195 - 1206. ISSN 0747-5632 (2009)

## **Cuadernia: una buena alternativa para la creación de actividades educativas**

Liliana Patricia Santacruz Valencia

Departamento de Lenguajes y Sistemas Informáticos,  
Grupo LITE. Universidad Rey Juan Carlos  
C/ Tulipán s/n, 28933 Móstoles, Madrid  
liliana.santacruz@urjc.es

**Abstract.** En un mundo cada vez más influenciado por la tecnología y en el que es importante proporcionar contenidos en lo posible interactivos a los estudiantes, es necesario contar con herramientas que permitan crear actividades educativas capaces de atraer a los alumnos mediante nuevas formas de interacción que enriquezcan el proceso de aprendizaje. En este artículo se ha seleccionado una de ellas, Cuadernia, de la cual se describen sus características y las posibilidades que ofrece para la creación de actividades educativas.

**Keywords:** Actividades educativas, interacción, nuevas tecnologías, cuadernos digitales, herramientas de autor.

### **1 Introducción**

Dentro del contexto educativo Internet ofrece una amplia gama de recursos entre los que se encuentran las herramientas de autor, las cuales permiten crear diversos tipos de contenido multimedia, proporcionando una forma de enriquecer los procesos de enseñanza y aprendizaje.

Algunas de estas herramientas son ampliamente conocidas y se utilizan para diseñar y crear actividades educativas. Es el caso, por ejemplo, de las populares JClick [1] y Hot Potatoes [2], las cuales a pesar de compartir la misma finalidad, la creación de actividades educativas, ofrecen entornos y procedimientos muy distintos para su utilización, lo que influye a la hora de elegir cuál de ellas utilizar.

Sin embargo, se puede decir que las herramientas para la creación de actividades educativas han evolucionado. Ahora cuentan con interfaces más amigables e intuitivas, que permiten al profesor crear actividades en pocos pasos, con resultados realmente llamativos.

Es el caso de Cuadernia [3], una herramienta de uso libre que permite crear actividades educativas digitales, las cuales pueden ser presentadas tanto en formato de cuaderno digital, a través de la red o de forma impresa. La manera de trabajar con esta herramienta es muy flexible ya que se puede obtener desde la red, descargar una versión para tenerla en local en el ordenador y poder así trabajar sin conexión a la red o también es posible obtener una versión portable para utilizarla mediante un dispositivo USB.

Cuadernia, es un proyecto realizado por la Junta de Comunidades de Castilla la Mancha y que la Consejería de Educación y Ciencia de Castilla la Mancha pone a disposición de los usuarios. Orientada a la creación de actividades educativas en forma de cuaderno digital. Una de las características que más llama la atención es su facilidad de uso, ya que cuenta con una interfaz de usuario visual que es sencilla e intuitiva, a través de la cual es guiado en el proceso de creación. Dicho proceso se desarrolla en un tiempo bastante corto después del cual se obtiene un producto final de gran calidad.

A continuación, en la sección 2 se describen las características más relevantes de Cuadernia, en la sección 3 se presenta un análisis comparativo de Cuadernia frente a otras herramientas disponibles en Internet con finalidad similar. En la sección 4 se plantea una breve reflexión sobre los aspectos a tener en cuenta en la selección de herramientas para la creación de actividades educativas y finalmente, en la sección 5 se presentan las conclusiones del tema abordado.

## **2 El entorno Cuadernia**

En esta sección se describe el entorno Cuadernia, para comprender cómo se lleva a cabo el proceso creativo, para qué sirve esta herramienta y cuál es la filosofía de trabajo propuesta.

## 2.1 ¿Para qué sirve Cuadernia?

Cuadernia permite la creación de contenido el cual se presenta en forma de páginas de un libro, representando perfectamente a una unidad didáctica formada por conceptos y ejercicios interactivos, que permiten evaluar los conocimientos adquiridos por el alumno. Así, el profesor puede recibir mediante correo electrónico las actividades resueltas por los alumnos para su correspondiente evaluación.

Un aspecto relevante a la hora de trabajar con Cuadernia es que cumple con los estándares internacionales para la creación de objetos de aprendizaje (contenido), los cuales se pueden adaptar a las necesidades de los alumnos y además son accesibles a toda la comunidad educativa a través de un repositorio de objetos de aprendizaje.

Los beneficios del uso de Cuadernia en el aula desde la perspectiva del alumno se traducen en que: *(i)* se siente más motivado, *(ii)* puede acceder al contenido a través de la web, desde cualquier dispositivo móvil e independientemente del sistema operativo utilizado, *(iii)* puede obtener el resultado de sus evaluaciones de forma automática y con las correcciones correspondientes, *(iv)* puede realizar una recopilación del material escolar de forma organizada, *(v)* puede comprender más fácilmente conceptos abstractos gracias a los elementos multimedia incluidos en las actividades *(vi)* se involucra más en el proceso de aprendizaje gracias a que las actividades fomentan la interactividad.

Desde el punto del profesor los beneficios son: *(i)* el desarrollo de actividades educativas se realiza a través de un proceso asistido, sencillo e intuitivo, *(ii)* el tiempo de desarrollo se reduce, *(iii)* el resultado de las evaluaciones realizadas a los alumnos se pueden recibir vía correo electrónico y son corregidas automáticamente, lo que se traduce en tiempo disponible para realizar otras actividades docentes, *(iv)* la participación colaborativa de los alumnos en las actividades propuestas se incrementa notablemente, *(v)* la disponibilidad de un repositorio de objetos de aprendizaje facilita la reutilización de contenidos en la creación de nuevos cuadernos digitales, *(vi)* el uso de objetos tridimensionales de realidad aumentada añade interactividad al contenido presentado a los alumnos, logrando captar su atención e incrementando la motivación en el aprendizaje, *(vii)* la disponibilidad de una comunidad de desarrolladores favorece la participación y comunicación entre docentes quienes pueden compartir su experiencia de uso de la herramienta.

## 2.2 El proceso creativo

Cuadernia es un maquetador de cuadernos y actividades, cuya interfaz gráfica se basa en ventanas, de la cual el usuario puede configurar ocho de sus características (ver Fig. 1).



Fig. 7. Configuración de características en Cuadernia.

Así, en el espacio de trabajo de la herramienta se identifican cinco ventanas, como se muestra en la Fig. 2.



Fig. 2. Interfaz gráfica de Cuadernia basada en ventanas.

- **Ventana Archivo:** Permite crear, abrir, guardar, cerrar o borrar un cuaderno, y también acceder al maquetado SCORM a través de un asistente, para su posterior incorporación a Gestores de formación online como Moodle.
- **Ventana Administrador:** Desde ella se puede acceder a operaciones para gestionar la galería de recursos, previsualizar un cuaderno y publicarlo.
- **Ventana de Edición:** Permite cambiar el fondo de las páginas, utilizar una cuadrícula de apoyo para el diseño de contenidos, copiar y pegar elementos y alinearlos en la página.
- **Ventana de Objetos:** Contiene botones que permiten insertar elementos multimedia dentro de la página que se está trabajando y escenas de realidad aumentada y de Descartes.
- **Ventana Página:** Permite gestionar el orden de las páginas del cuaderno y navegar a través de ellas.

El proceso creativo con Cuadernia se basa en tres pasos muy sencillos: (i) abrir el entorno, (ii) crear o abrir un cuaderno y (iii) crear contenidos y actividades, como se muestra en la Fig. 3.

Al abrir el entorno, el usuario se encuentra con una interfaz amigable muy fácil de utilizar, puesto que su manejo es bastante intuitivo, por lo tanto, el profesor puede empezar a diseñar su cuaderno digital de una forma rápida y sencilla.

Si ya se ha realizado un cuaderno digital, al abrir el entorno el usuario puede verlo y realizar modificaciones sobre el mismo, si así lo desea.

Pero si lo que el usuario quiere es crear un nuevo cuaderno dispone para ello de una gran variedad de elementos multimedia que puede utilizar para incluirlos en las actividades (Cuadernia dispone de un total de 16 actividades evaluables), que forman parte del mismo.

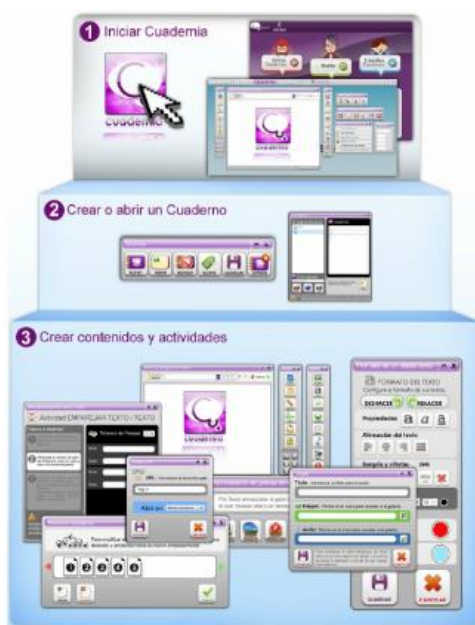


Fig. 3. Proceso creativo con Cuadernia.

### 2.3 Filosofía de trabajo

Al crear un cuaderno digital el usuario puede incluir cuantas páginas considere necesarias, constituidas por diferentes objetos multimedia de diversa naturaleza como texto, imágenes estáticas y animadas, sonidos, animaciones flash, actividades interactivas, enlaces a otros recursos y objetos de realidad aumentada. Además, el



usuario puede incluir actividades interactivas que permiten a los alumnos afianzar los conceptos adquiridos.

Finalmente, el cuaderno resultante se puede exportar a una archivo .ZIP para su posterior distribución y/o almacenamiento. Esta metodología de trabajo se ilustra en la Fig. 4.

Cuadernia cuenta con una excelente documentación, que facilita el aprendizaje de las distintas funcionalidades del entorno.



Fig. 4. Filosofía de trabajo con Cuadernia.

### 3 Comparativa con otras herramientas

La web ofrece un abanico de posibilidades a la hora de elegir una herramienta para la creación de actividades educativas. Dentro de ese gran espacio de selección se han seleccionado herramientas como Tikatok [4], que permite la creación de actividades educativas utilizando plantillas predefinidas que sirven para desarrollar una historia o un proyecto adecuado a las necesidades de la clase, e por tanto una herramienta más orientada a los estudiantes.

Otras herramientas como LetterPop [5], permiten crear boletines electrónicos simplemente arrastrando y pegando el texto y las imágenes, las cuales pueden ser incluidas desde Flickr o utilizando las que proporciona la propia herramienta.

Pancho y su máquina de hacer cuentos [6], es una herramienta orientada a la creación de cuentos mediante el uso de plantillas. Permite incluir imágenes, música y texto.

Calaméo [7], permite crear revistas digitales y también trabajar con diferentes formatos de texto, proporcionando facilidades de conversión de formatos como PDF, Word, PPT, OpenOffice entre otros.

Isuu [8], es otra herramienta que facilita la creación de revistas digitales en formato Flash, con texto e imágenes y que pueden ser vistas en online, con posibilidad de hacer zoom. Se pueden distribuir a través de las redes sociales como Facebook, Google+ y LinkedIn.

También es posible crear cuadernos digitales con fotos, tarjetas y calendarios con herramientas como Mixbook [9].

Si lo que se quiere es crear un libro con imágenes y texto acompañados de una voz sintetizada se puede utilizar Tar Heel Reade [10], es multilingüe y permite crear cuentos y distribuirlo en formato PPT, Flash o impreso.

Otra forma de crear libros virtuales es utilizando ClassTool [11]. Se pueden distribuir a través de Wikis, Blogs o páginas web y el resultado final se puede imprimir.

Ardora [12], es una herramienta que genera automáticamente páginas web. Ofrece cuarenta y cinco tipos de actividades educativas, plantillas multimedia y posibilidad de incluir páginas para comentarios, chats, anotaciones.

Sin duda, una de las más conocidas es JClic [1], una herramienta que cuenta con un applet para insrustar las actividades JClic en una página web (JClic applet), un visor (JClic Player), una herramienta de autor (JClic Author) y una herramienta para llevar a cabo estadísticas (JClic Report). Es un entorno muy completo para crear actividades

educativas, cuyo producto final se obtiene en una carpeta .ZIP que contiene todos los recursos multimedia utilizados en la creación de las actividades educativas, que pueden ser del tipo sopa de letras, puzzle, asociaciones, completar, entre otras. Desarrollada en lenguaje Java, almacena los datos de las actividades en formato XML, lo cual permite su integración en bancos de recursos de estructura compleja y la reutilización de proyectos JClic en otras aplicaciones. Es una herramienta multiplataforma que soporta diferentes formatos de imágenes (BMP, GIF, JPG y PNG), de audio y vídeo (WAV, MP3, AVI, MPEG) y soporta QuickTime y Flash. Además es un proyecto de software libre.

Y entre las conocidas está Hot Potatoes [2], una herramienta que utiliza JavaScript para la creación de actividades educativas interactivas, mediante el uso de un conjunto de herramientas especializadas en la creación de ejercicios para responder diversos tipos de preguntas (JQuiz), ejercicios para rellenar huecos (JCloze), ejercicios de crucigramas (JCross), ejercicios para ordenar palabras, oraciones, textos (JMix), ejercicios para relacionar conceptos (JMatch) y una herramienta para enlazar todos los ejercicios creados (The Masher). Además es posible configurar el idioma de la interfaz de usuario. Las actividades creadas se pueden difundir fácilmente a través de la web. Es necesario registrarse para utilizar la herramienta, y la versión comercial proporciona funcionalidad completa para The Masher.

**Tabla 1.** Comparativa entre herramientas de autor para creación de actividades educativas.

	Acceso		Entorno trabajo		Interfaz		Proceso creativo		Tipo de Actividad		Producto final				Difusión		Usuario		SCORM		
	Registro online gratuito	No registro	Online	Descarga	Página web	Ventanas	Plantillas	Asistido	Actividades (Puzzles, etc.)	Cuentos, otras	Cuaderno Dig.	Página web	Cuento	Carpeta .ZIP	Embebido web	Carpeta .ZIP	Multiformato o PPT y/o Flash	Redes Sociales		Profesor	Alumno
Tikatok	✓				✓						✓										
Letterpop	✓		✓		✓		✓			✓											
Pancho		✓	✓		✓		✓			✓			✓		✓				✓		
Calaméo		✓	✓		✓					✓											
Isuu	✓		✓		✓			✓		✓					✓				✓		✓
Mixbook	✓				✓			✓		✓					✓				✓		✓
Tar Heel Reader	✓		✓		✓			✓		✓					✓				✓		✓
ClassTool		✓	✓		✓			✓		✓		✓			✓				✓		
Ardora		✓	✓		✓		✓			✓		✓			✓				✓		✓
JClic		✓		✓	✓	✓		✓	✓				✓		✓				✓		
Hot Potatoes	✓		✓		✓			✓	✓			✓			✓				✓		
Cuadernia		✓		✓	✓	✓		✓	✓		✓				✓				✓		✓

## 4 Reflexión

Cuando se trata de elegir una herramienta para el desarrollo de actividades educativas, es importante identificar ¿Por qué? y ¿Cuándo? el uso de dichas herramientas puede enriquecer el proceso de enseñanza aprendizaje.

Al analizar el ¿Por qué? se debe tener en cuenta que: *(i)* el estudiante es muy social, le gusta formar parte de comunidades. Desea tener una realimentación rápida y un aprendizaje personalizado. Es muy visual y necesita actividades que le motiven, es el llamado nativo tecnológico y eso es bueno, puesto que está en cierta forma preparado para aprender los contenidos, los conceptos y adquirir competencias a través de las herramientas que el profesor le proporcione. Otro elemento a tener en cuenta es *(ii)* el aprendizaje, las teorías de aprendizaje están muy en línea con el buen uso de la tecnología, algunas son el aprendizaje social, el aprendizaje colaborativo, el aprendizaje activo o el aprendizaje social constructivista. Todas ellas involucran al estudiante para que participe en actividades en grupo muy y fomentan el desarrollo del conocimiento colectivo. Y el otro componente es *(iii)* el programa curricular, es necesario contar con herramientas que permitan abordar el programa curricular de forma que los estudiantes dispongan de una variedad de experiencias de aprendizaje que les ayuden a construir su propio conocimiento, a través de la colaboración y la comunicación, de una forma diferente a como se puede plantear en la educación tradicional.

Para identificar el ¿Cuándo? Hay que pensar en diferentes oportunidades, por ejemplo *(i)* cuando se quiere proporcionar acceso a algo más allá del aula, probablemente un entorno virtual, por ejemplo, en el cual los estudiantes puedan encontrar oportunidades para interactuar y poner en práctica los conceptos aprendidos en el mundo real. Otra oportunidad es *(ii)* cuando se desea mejorar los resultados de las evaluaciones, ya que mediante el uso de elementos multimedia soportados por muchas de las herramientas para la creación de actividades educativas, los conceptos abstractos se pueden presentar al os estudiantes de forma más interactiva, facilitando su comprensión. Y finalmente, se utilizan estas herramientas, *(iii)* cuando se quiere proporcionar oportunidades para una mejora continua del aprendizaje, apoyada en la realimentación, la corrección de errores y la generación de respuestas de forma automática.

Una forma de facilitar la elección de una herramienta para la creación de actividades educativas, dentro de la amplia gama existente en la red puede ser intentar dar respuesta a las siguientes preguntas: Esta herramienta proporciona a los estudiantes:

- ¿Acceso a actividades educativas que de otra forma no podrían utilizar?
- ¿Oportunidades para interactuar con otros alumnos en el desarrollo de proyectos u otra clase de actividades educativas?
- ¿Formas mejoradas para aprender conceptos abstractos?
- ¿Mecanismos para recibir realimentación en una forma más efectiva y rápida?
- ¿Personalizar experiencias de aprendizaje que estén en línea con sus estilos de aprendizaje?
- ¿Experiencias que los preparen para actividades que deberán realizar en su trabajo profesional y en el mundo real?
- ¿Oportunidades de autoevaluación que les permitan reflexionar sobre lo que saben y lo que no?
- ¿Oportunidades para hacer un seguimiento de su progreso en la experiencia de aprendizaje?

Por tanto, si a la hora de escoger la herramienta que se quiere emplear para la creación de actividades educativas, la mayoría de las preguntas planteadas son respondidas afirmativamente, significa que la herramienta que se quiere elegir es la adecuada, de lo contrario, probablemente no lo sea.

Si se aplica esas preguntas a una herramienta como Cuadernia, se puede comprobar que la mayoría de ellas se responden positivamente, ya que en el caso querer realizar el seguimiento del progreso del estudiante, probablemente la elección acertada sería JClic, de entre las herramientas analizadas.

En cualquier caso, el listado de preguntas es orientativo. Un ejercicio propuesto al docente sería el de realizar un listado de preguntas, teniendo en cuenta cuál es el objetivo pedagógico planteado y las características de los alumnos que van a realizar las actividades creadas con una herramienta en concreto, de esa forma la elección de se acercará más a lo que se busca.

## 5 Conclusiones

En este artículo se han descrito las principales características de la herramienta para la creación de actividades educativas Cuadernia. También se ha resaltado su facilidad de uso, gracias a un entorno visual intuitivo que permite la creación de dichas actividades de forma rápida y mediante la incorporación a las actividades de elementos multimedia y objetos tridimensionales de realidad aumentada. De otra parte, se han identificado las bondades de utilizar Cuadernia en el aula, tanto desde el punto de vista del estudiante como del profesor.

Se ha comparado Cuadernia con otras herramientas disponibles en la web y que ofrecen posibilidades similares para la creación de actividades educativas.

Para finalizar el artículo se ha planteado una reflexión a través del análisis distintos aspectos a tener en cuenta a la hora de seleccionar una herramienta para la creación de actividades educativas y se ha ofrece una lista de preguntas que podrían servir para facilitar el proceso de selección, teniendo como centro lo que esa herramienta puede aportar al estudiante.

## Referencias

- [1] JClíc. Disponible en <http://clic.xtec.cat/en/jelic/>. Último acceso 5 de diciembre 2013.
- [2] Hot Potatoes. Disponible en <http://hotpot.uvic.ca/>. Último acceso 5 de diciembre 2013.
- [3] Manual de Cuadernia. Disponible en:
- [4] [http://repositorio.educa.jccm.es/portal/Cuadernia/manual\\_cuadernia\\_v3.pdf](http://repositorio.educa.jccm.es/portal/Cuadernia/manual_cuadernia_v3.pdf)
- [5] Último acceso 5 de diciembre 2013.
- [6] Tikatok. Disponible en <https://www.tikatok.com/>. Último acceso 5 de diciembre 2013.
- [7] Letterpop. Disponible en <http://www.letterpop.com/>. Último acceso 5 de diciembre de 2013.
- [8] Pancho y su máquina de hacer cuentos. Disponible en:
- [9] [http://portal.perueduca.edu.pe/modulos/m\\_pancho/](http://portal.perueduca.edu.pe/modulos/m_pancho/)
- [10] Último acceso 5 de diciembre 2013.
- [11] Calaméo. Disponible en <http://es.calameo.com/>. Último acceso 5 de diciembre 2013.
- [12] Isuu. Disponible en <http://issuu.com/>. Último acceso 5 de diciembre 2013.
- [13] Mixbook. Disponible en <http://www.mixbook.com/>. Último acceso 5 de diciembre 2013.

- [14] Tar Heel Reader. Disponible en <http://tarheelreader.org/>. Último acceso 5 de diciembre 2013.
- [15] ClassTool. Disponible en <http://www.classtools.net/>. Último acceso 5 de diciembre 2013.
- [16] Ardora. Disponible en <http://webardora.net/>. Último acceso 5 de diciembre 2013.

## Recommendations on the Design of Semi-Immersive Interactive Computer Educational Applications for Children

Haydeé Artaza Alvarez, Diana Perez Marín

Languages and Systems I department, Rey Juan Carlos University,  
Tulipán street, 28933, Móstoles, Madrid  
{hartaza, diana.perez}@urjc.es

**Abstract.** In the last decades, computers have been used for education too. Several benefits of using computers for education have been studied. However, there are still some Human-Computer Interaction design issues on how to create new semi-immersive interactive computer applications for education. In this paper, the focus is on helping any Computer Engineer that would like to create a semi-immersive computer educational application for children. Semi-immersive computer applications can create the feeling on the users that they are living the content provided. In particular, for education, semi-immersive applications can have a great potential because they involve more senses and thus, students are supposed to learn better and remember the content longer. We have created one of these systems and 41 students have used it for a course. This experience has provided us with a set of recommendations for the design of semi-immersive interactive computer educational applications for children such as the need of combining music, customized animated pedagogic agents and scenario-based stories to achieve a high satisfaction of the students and to improve their long-term retention of the content of the system.

**Keywords:** Computer-supported education, Immersive computer application, Human-Computer Interaction, Software Design.



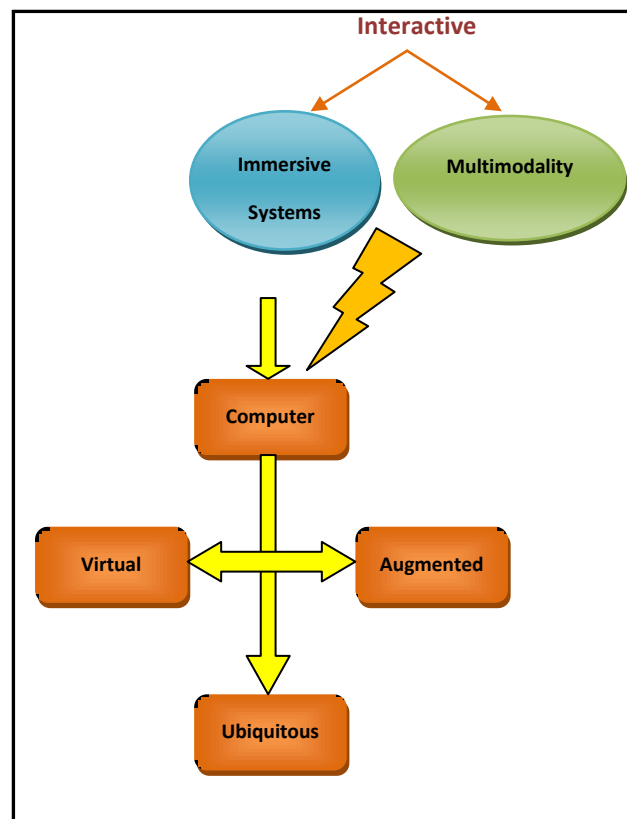
## 1 Introduction

Computer systems should provide an intuitive and simple interface to facilitate user interaction. The discipline known as Human Computer Interaction (HCI) provides developers with guidelines for the design, evaluation and implementation of interactive computing systems for human use [1]. This discipline proposes the development of safe, effective, usable, and accessible interactive systems.

Figure 1 shows the four paradigms or models from which all interactive systems are derived according to HCI. These paradigms are the following:

- **Computer desktop.** The user interacts with a computer by direct manipulation.
- **Virtual reality.** A virtual world is created in the computer, so that the user feels direct physical presence, sensory cues in three dimensions, and natural interaction.
- **Ubiquitous computing.** The computer is relegated to the background, trying to be "transparent" to the user.
- **Augmented reality.** The physical world is enriched with digital information that the user can see mixed with the real world.

This paper focuses on immersive, multimodal interactive systems situated within the computer desktop paradigm. This is because it is currently the dominant paradigm due to its low cost (see Fig. 1). Immersive systems are a subset of interactive systems [2]. Immersion is an extension of the technique known as information hiding [3]. In this way, the user does not perceive that you are interacting with the computer or other device, which entails a natural interaction with the system. On the other hand, multimodality refers to offer the content in several communication channels involving different senses such as sight, hearing, smell or taste.



**Fig. 1.** HCI paradigms of interactive systems. This paper focuses on immersive, multimodal interactive systems situated within the computer desktop paradigm.

There are various levels of immersion provided in an immersive system [3]:

- **No acceptance.** Users know that the program is launched and runs in a computer.
- **Low acceptance.** Users find some immersive features of the environment but they do not get involved with them.
- **Acceptance.** Users focus on the interaction with the virtual world created by the computer, but they are still fully capable of distinguishing between the two worlds.

- **Total mental immersion.** Users feel fully immersed in the virtual world. They are detached from the real world.

It is advisable to choose a level of immersion with greater or lesser depth depending on the need for realism, the availability of technology and the cost. For instance, in our case, we chose low acceptance with the design of a semi-immersive computer educational application because we wanted to use the system in Spanish schools in which students do not have sensors, helmets or highly expensive technology. Moreover, we have decided to use music, and animated pedagogical agents, i.e. interactive characters that talk to the users to make a more natural, friendlier and easier to use interface.

41 students have used the developed system. This experience has provided us with insightful information from which we have devised a set of recommendations for any designer that wants to create semi-immersive educational applications.

This paper is organized as follows: Section 2 presents the state of the art review in immersive systems; Section 3 describes the devised semi-immersive computer educational application; Section 4 focuses on the experiment performed and the results achieved; and, finally Section 5 concludes the paper with the recommendations.

## 2 State of the Art Review

### 2.1 Context

Interactive systems are characterized by a significant amount of interaction between man and computer [1]. These systems have evolved in recent decades from relying exclusively on the command line interaction (e.g. the DOS operating system), to sophisticated graphical interfaces based on the use of menus (e.g. Windows 7 operating system). According to [2] interactive systems can be classified as follows:

- **Command line systems.** Systems controlled by command line. Users must learn and remember the commands and their arguments.
- **Graphical systems.** Systems that provide the user with a graphical interface as menu and dialog systems, using the keyboard as the main input device and with numbered set of options.

- **GUI systems.** Systems based on the desktop metaphor that provide users with graphical menus in which the options can be selected by using a mouse pointer and clicking on graphical icons.
- **Virtual systems.** Users explore a virtual world with little or no symbolic translation. Visualization techniques and new input devices like data gloves are used.
- **Agent-integrated systems.** Users interact with the systems by using avatars which can talk to them and select options of the applications.
- **Embedded systems.** Systems that perform dedicated functions in real time and on their own.

**Immersive systems** can be subclassified as one type of virtual interactive systems, which can be ideal for training [4]. As indicated in the Introduction, these systems can show a continuum of degree of immersion from no immersion to total mental immersion [5]. However, total mental immersion does not guarantee a higher efficiency. Cognitive realism is more important, that is, to create a simulated environment that provides real experiences [4].

For instance, in the study by [6], which investigates the use of immersive environments for learning mathematical sign language for deaf children, despite using a total mental immersion system they had many problems. In particular, they found many limitations such as the minimum number of hand shapes (inputs) recognized by the system, the high cost of equipment, health issues and safety issues associated with the use of immersion devices.

On the other hand, **semi-immersive systems** can be a good alternative when the user is required to keep in touch with real world elements. Semi-immersive systems still provide a greater sense of presence than non-immersive systems, and they are an attractive option with little investment. They are currently used as part of military training, tools in the industry, medicine, and education showing great advantages [4,6-8].

## 2.2 Technologies

Users can interact with (semi)-immersive systems, reacting spontaneously, so that they acquire awareness of the consequences that their actions can cause. The environment can express their reactions through sounds, smells, etc.

According to the study [8], in the design of an immersive environment, a technique known as mixed reality integrates real world components relying on interactive digital media such as videos. **Mixed reality** combines traditional computer interfaces with visual and auditory **simulations**. According to this technique, a team of researchers, teachers, psychologists, computer scientists and artists have to work together to create the environment.

Fassbender et al. [9] also highlight the importance of musicians and the benefits of music in immersive learning systems. The type of music (audio) to be included in the environment should be identified to make it more attractive, fostering motivation and, in the case of education, promoting student learning.

Another technique to enrich the immersive environments is the inclusion of **animated pedagogical agents**, animated characters to guide users within the environment, sometimes even using **natural language** processing techniques [7].

For instance, Self [10], Prevost et al. [11], and Nunes et al. [7] propose the creation of **customized pedagogic agents** to adapt themselves to each student's cognitive ability, and personal features. Paulus et al. [12] argue that agents can also provide a context to the simulated environment, recreating a story in which the student has to complete several activities to achieve one or more goals. This perspective suggests that the student, as a kind of player or protagonist, takes part of a story related to the topic you want to learn. In this way, the student perceives the activity of studying and learning as fun and entertaining.

Reategui et al. [13] propose an approach that combines the use of animated pedagogical agents and stories using the simulated environment as context to enhance the learning process. In particular, Reategui et al. defined the concept SBeL defined by its initials in English as "**Scenario-based e-learning language**", in which the use of activities to improve communication skills in e-learning environments is proposed. Through SBeL, a story with elements such as video, audio and animation can be defined. Next, students have to interact with the environment by making a series of tasks or activities helped by the guidance of pedagogic agents which talk to them in each scenario.

The affinity and the bond that the agent can create with the student is also important [7,10]. Finally, the study by Arroyo et al. [14] shows the results of an experiment in which the effect of the gender and the human form of the agent was analyzed. They found that girl students that chose a man agent achieved better results than girl students that chose the woman agent.

### 3 The Iberian Lynx System

From the state-of-the-art review, it can be concluded that the use of semi-immersive environments can be beneficial for education. Moreover, it has been found out some interesting techniques to design educational semi-immersive environments such as mixed reality, music, customized animated pedagogic agents and scenario-based stories.

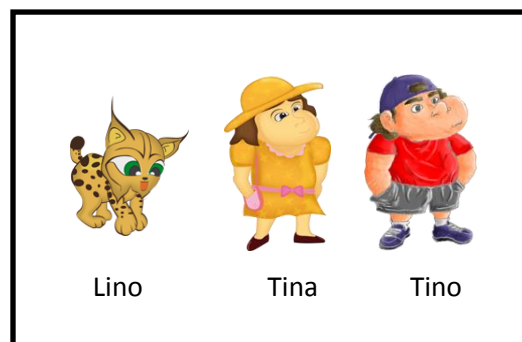
In this section, the development of a semi-immersive educational application using the previous techniques is described. In particular, the application called ‘The Iberian Lynx’ is a realistic environment that simulates the ecosystem of a particular endangered species.

The student can interact with the environment as part of the simulated ecosystem to learn about the needs and problems of the Iberian Lynx. In the following subsections, the features and technical specifications of the system will be described.

#### 3.1 Features

- **Semi-immersive system.** Intended to provide a more realistic environment than real, i.e. "cognitive realism" (Herrington, 2007) through student interaction with the environment and environmental responses through messages. The main objective of the system is to show the child the problem of extinction of the Iberian Lynx. Through interaction with the characters in the story the child learns the habits of Lynx on the environment, and the problem faced by this species due to some habits of human beings. Thus, it is likely that the student becomes aware of the problem, acquiring an aptitude and ecological behavior.

- **Animated pedagogical agents.** Represented by two avatars, one that personifies the student within the environment, and the second avatar that plays the role of guide into all scenarios (Nunes et al., 2002). Each avatar provides knowledge and guidance to the student throughout the visit by environment. The following avatars were created (see Fig. 2):
  - Boy student avatar: Tino.
  - Girl student avatar: Tina.
  - Baby Iberian Lynx avatar: Lino.



**Fig. 2.** Animated Pedagogical Agents. Children can choose male or female agent.

The reason to design two different, girl and boy, avatars for students was to analyze the bond that can be created between the student and the avatar according to its gender [14].

- **Multimodal interaction.** Interaction with the environment through input devices (keyboard, mouse), including digital media such as video and audio, appropriate to each scene of environment to encourage learning [8,9]. The sounds present in each scene are significant to place students within the scene. Besides, the characters that make up each scene can communicate with the student through dialogues represented in bubbles.
- **History divided into scenes.** Development of an environment based on a story as proposed by [13]. The goal is that the student learns by participating and playing. At each scene the student performs a series of interactive activities, guided by animated pedagogical agent, which

facilitates the learning of environment. Students can move between the scenes of the story having full control over the environment. The story is divided into the following scenes:

- **Presentation:** the agent presents itself to the student.
- **Introduction to Iberian Lynx habitat:** the agent shows the student the Iberian Lynx habitat.
- **Description of Iberian Lynx habitat:** the agent explains to students the habits and activities of the Iberian Lynx.
- **Problems that affect the conservation of the species:** the agent explains to the student the problems facing the Iberian Lynx concerning the protection of the species.
- **Farewell / End:** the agent dismisses the student and expresses the need to disseminate what s/he has seen and learnt with their friends so they know the problem and can also help to solve it.

### 3.2 Technical specifications

The e-Adventure software has been used to create the environment. It is a platform for the development of classic adventure games for educational purposes. This platform is free software under the terms of the GNU Lesser General Public License, owned by the University <e-UCM> Complutense de Madrid. e-Adventure supports several versions of Windows (XP, Vista and 7) and Mac OS X (Leopard and snowleopard) and Linux. It uses Java version 1.6.0\_18, and it has been used in its version v1.2 RC2.



## 4 Experiment

### 4.1 Motivation and research questions

The aim of the experiment was to reflect on how semi-immersive systems can be designed to achieve educational goals such as, in this case, that students become aware of the problem of the Iberian Lynx. Other related research questions addressed were the following:

- To what extent can the gender of the participant influence in the decision of choosing a girl/boy avatar in the immersive system?
- To what extent can the interaction with the immersive system influence into the perception and the concept that students have about the animals?
- Does the use of a multimodal immersive learning system improve the learning ability of children of school age?
- Does the use of a multimodal immersive learning system improve environmental awareness among school children?
- Do children prefer to use the immersive system or just read a leaflet to learn about the Iberian Lynx?
- Does the immersive system have a greater impact on the retention of long-term content compared to the activity of reading the leaflet?

### 4.2 Participants and used tools

The sample for the experiment consisted of 41 school children between 13 and 15 years old. They were all from a public school in Madrid (Spain), and they participated anonymously and voluntarily in the experiment. Neither the teachers nor the students had previously had contact with immersive multi-modal systems, although they were familiar with the use of interactive computer systems.

The tools consisted of the semi-immersive system, educational leaflets (see Figure 3) and questionnaires. Both groups of children were randomly divided into two subgroups (although the similarity between the students of the two subgroups was checked):

- **Control group:** 19 children using the semi-immersive computer educational tool.

- **Test Group:** 22 children using the leaflets.

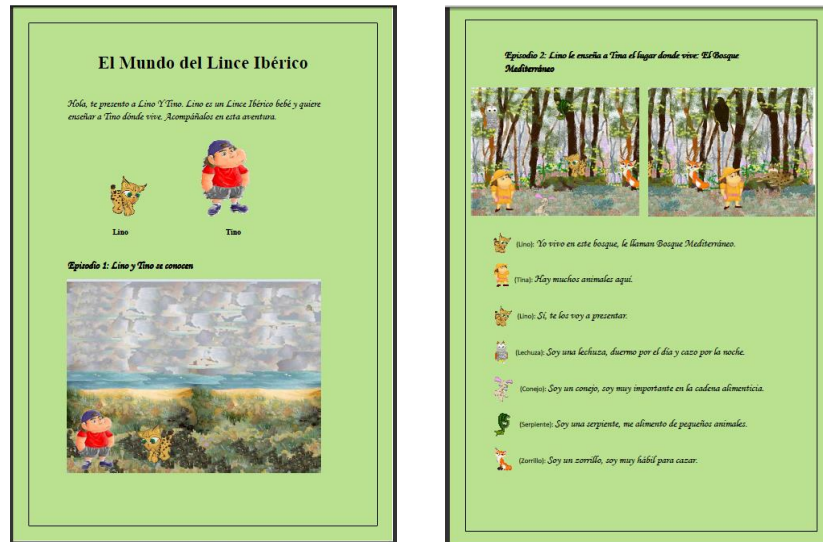


Fig. 3. Educational leaflets used in the test group. These leaflets show the Iberian Lynx situation.

### 4.3 Development of the experiment

The experiment was organized in the following phases:

- Previous survey (pre-test phase) to control group and test group, about knowledge and ecological awareness on the Iberian Lynx. These questionnaires are associated with each participant with an identification number, so the identity remains anonymous to each student.
- The test group, during a class, read a leaflet with information on the Iberian Lynx. Students could choose between two types of leaflets, one with the main character represented by a boy, and another with a girl character.
- In the computer room, the control group, during a class, used the semi-immersive environment that shows the same ecological issue presented to the test group. At the beginning of the experiment, students were given a brief explanation on how to use the system and that they could choose between two versions of the protagonist: a girl avatar or a

boy avatar. Besides the freedom of choosing the character, students were free to interact with the system.

- Final survey (post-test phase) in both groups of students to evaluate changes in knowledge and ecological awareness on the Iberian Lynx. Besides the survey, both groups wrote a brief summary of the activity, and a brief personal view of the activity.
- After two months, it was asked to each group to complete a new survey about the problem outlined in each activity with questions such as: do you remember the problem about lynxes and how could it be solved? This approach is intended to measure the effect of the computer system versus the paper leaflets on the long-term memory of each group.

Finally, the results were compared between the control and test groups to draw some conclusions.

#### **4.4 Results and discussion**

The results obtained in the questionnaires have provided us with a qualitative and quantitative analysis of the effects of using the semi-immersive system with the students. The questions of the questionnaires are divided into three groups:

##### **General questions:**

- Q1: Are you a girl or a boy?
- Q2: How old are you?

##### **Open-ended questions:**

- Q3: Do you like animals? Why?
- Q10: Do you think that the Iberian Lynx is an endangered species? Why?
- Q11: If the Iberian Lynx were an endangered species, how can you help you to make this situation change?

##### **Multiple-choice questions:**

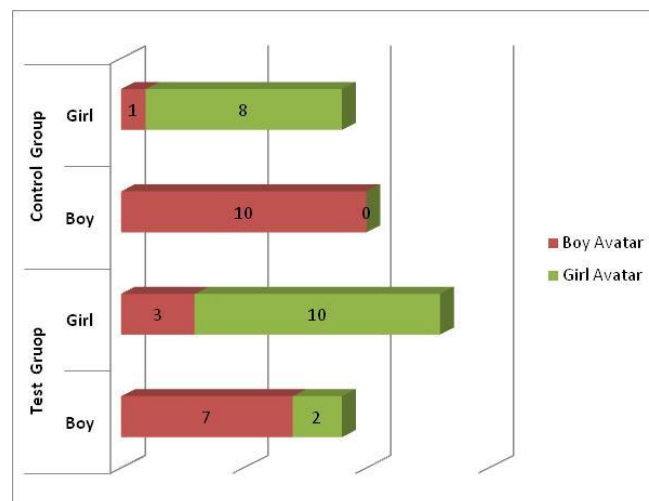
- Q4: What is the Iberian Lynx?
- Q5: Do you know or imagine where the Iberian Lynx lives?

- Q6: Do you know what feeds the Iberian Lynx?
- Q7: Do you know how many babies can have a female Iberian Lynx?
- Q8: What does it mean for you the term "ENDANGERED SPECIES"?
- Q9: Why do you think that some animal species are endangered?

The analysis of the results of the questionnaires has provided relevant information that has made possible to analyze each major points of this research, and discuss the results in the following subsections.

#### *Relationship between the student / avatar gender*

Both in the control group and the test group students have preferences for avatars of the same gender (see Fig. 4). These results suggest that there is an affinity with the avatars of the same gender that the student.



**Fig. 4.** Avatar gender distribution according to participants. In general the students prefer avatars of the same gender.

In the control group all the children have chosen avatars of the same gender. In the case of girls, only one student chose an avatar of the opposite sex. In the test group 9.09% of children (2 boys) have chosen an avatar of the opposite sex, and girls for the 13.66% (3 girls) have chosen a child avatar.

### *Influence on the perception of animals*

According to the results gathered to Q3, in the pre-test phase, all students in general consider the animals as being tender and loving, providing companionship. This relationship is perhaps due to the influence of human-pet relationship that is usual for them. All the same, answers such as "Part of the nature" and "Knowing them is interesting / Diversity" have also been found in both groups. It could indicate that these students have a different concept of animals, that is, they consider them more than just a pet.

In the post-test phase, for both groups, there are also concepts like "Care for and protect", "Respectable beings" and "Different from humans". These new concepts show the influence of activities which has permitted the extension of the concept that children had about animals, that is, they see now the animals more than just pets.

### *Effects on learning ability*

The results of Q4-10 of the questionnaire are used to measure the improvement in the learning. In all cases, the variable "Variation" represents the difference of values registered between the pre-test and post-test phases in each group measuring the positive or negative effect of using the semi-immersive computer educational system described in Section 3. Table 1 presents a summary of the variations obtained for questions 4-9 (see Section 4.4).

**Table 1.** Summary of variations for Q4-9.

Question	Control group	Test group
Q4-concept	0%	0%
Q5-place of living	21,05%	13,64%
Q6-food	36,84%	36,36%
Q7-children	52,63%	40,91%
Q8-endangered	0%	0%
Q9-reason	10,53%	4,55%
Average	30,26%	23,87%
Average variation	6,39%	

As can be seen, questions 4 and 8 have had a variation of 0% so no effect has been registered. Regarding to questions 5, 6, 7 and 9, several variations have been registered (although none of them significant). Nevertheless, these results could provide some evidence that the activity of interaction with the immersive system has been more effective than the activity of reading the leaflet.

Given that question 10 is a subjective question, a numerical value was assigned between 0 (no effect) to 5 (maximum effect, i.e. the student learns that the Iberian Lynx is an endangered species) to calculate the variation of answers before and after the use of the system. The results are shown in Table 2.

**Table 2.** Average variations registered for Q10.

Phase	Control group	Test group
Pre-Test	3,05	2,36
Post-Test	4,53	4,59
Average variation	+1,48	+2,23

As can be seen, the average variation was greater for the test group by 0.75 points.

#### *Effects on environmental awareness*

The effects on environmental awareness are analyzed from the results gathered to question 11, and a set of summaries collected from both groups. Given that these answers are subjective, a numerical value has been assigned to each response according to a scale between 0 (no effect) to 5 (maximum positive effect, i.e. the student is now able to understand the effects of the extinction of the lynx on the ecosystem). Table 3 gathered the average variation score obtained from the answers to question 11.

**Table 3.** Average variations registered for Q11.

Phase	Control Group	Test Group
Pre-Test	3,47	2,86
Post-Test	4,53	5
Average variation	+1,06	+2,14

As can be seen, the test group seems to have understood better the effects of the extinction of the lynx on the ecosystem. However, Table 4 gathers the results of the analysis of the scores for the summaries written by the students. The variable "Average score" represents the average score obtained from each group calculated as the quality of their summaries from 0 (no quality) up to 5 (perfect quality, i.e. excellent summary with all the relevant information).

**Table 4.** Average quality of the summaries gathered.

Results	Control Group	Test Group
Average score	3,68	3,31

As can be seen, the average quality of the summaries of the control group (no use of the system) seems to be slightly better (+0.37 points). With these conflicting results, it is not possible to draw any conclusion on the improvement on the children's environmental awareness as a result of the interaction with the semi-immersive system.

#### *Effects on long-term retention of the information*

Students were asked to write a summary one month later to check the effects of using the system on long-term retention of the information. Table 5 gathers the results of the percentage of students of control and test groups that were able to write an excellent summary (qualified with a 5 by the teachers) of the information provided about the Iberian Lynx.

**Table 5.** Percentage of students who remembered the information about the Lynx one month later.

Results	Control Group	Test Group
Percentage	45.45	63.16

As can be seen, more than half of the students of the test group (+17.71% with respect to the control group) were able to produce an excellent summary one month later of using the computer system.

## 5 Conclusions and future work

In this work, a study on the use of semi-immersive environments for multimodal learning has been undertaken. In order to do that, a semi-immersive multimodal system which allows students to interact with the environment aided by an Animated Pedagogical Agent has been developed.

In the system, students are represented by an avatar, which in turn is a character in the context of a story that shows the problem of extinction of the Iberian Lynx. 41 students between 13-15 years old have used the system split into control and test groups. After performing the experiment and the analysis of the data, the following recommendations for any designer, teacher or researcher that wants to work with semi-immersive environments for education can be concluded:

- It is not necessary to include expensive technology to develop a semi-immersive multimodal system which allows user interaction with the environment.
- The multimodality feature of the system is a key factor as children use multiple senses when interacting with the environment. Some comments registered were *"I have been struck by the sad Lynx"* or *"I like the background music"*.
- The use of Animated Pedagogical Agents to guide the simulated environment attracts the attention of the student.
- Students feel an affinity for agents of the same gender (89.47% of the control group and 86.36% of the test group have chosen avatars of their same gender). Therefore, it is important to give the students the option to choose the avatar of their own gender.
- Students enjoy using the system so it is possible to design interactive exercises with some kind of reward to keep their initial motivation. Some comments provided by the students were *"I like using the system"* or *"I think that this is a very educational activity"*.
- The use of semi-immersive educational systems could have a significant effect on the long-term knowledge retention. It is because 63.16% of the test group students were able to produce an excellent summary of their interaction with the environment one month after using the system, while only 45.45% of test group students that read a leaflet on the Iberian Lynx could produce such high quality summaries of their reading.



Some possible lines of future work are the following:

- To perform a larger experiment with more children grouped by several features such as gender, age, previous knowledge of the discussion topic and competence using the computer to study the influence of these factors on the design of immersive systems.
- To include the possibility of allowing the children to adapt the environment as they wish and study the effect of this change in the results gathered on Q1-11.
- To include the possibility that children can make their own questions and choose between more than one story to analyze if that improves the learning.
- To compare the results achieved when the degree of immersion of the quality of the graphics and sounds are increased taking into account the cost of those changes.

**Acknowledgements.** We would like to thank the teachers and students of the School for giving us the possibility of going to class with them.

## References

- [1] Abascal, J. & Moriyón, R. (2002), 'Tendencias en Interacción Persona Computador', *Revista Iberoamericana de Inteligencia Artificial*(16), 9-24.
- [2] Spring, M. B. (2002), 'Interactive Systems', *Computer Sciences*. 2002. Encyclopedia.com, <http://www.encyclopedia.com>.
- [3] Nielsen, J. ed. (1993), *Usability Engineering*, Cambridge.
- [4] Herrington, J. (2007), 'Immersive Learning Technologies: Realism and Online Authentic Learning', *Journal of Computing in Higher Education* 19(1), 88-99.
- [5] Costello, P. (1997), 'Health and Safety Issues associated with Virtual Reality-A Review of Current Literature', *Loughborough University*, 6-9.
- [6] Adamo-Villani, N.; Carpenter, E. & Arns, L. (2006), 'An immersive virtual environment for learning sign language mathematics', *SIGGRAPH '06*.
- [7] Nunes, M. ; Dihl, L. L.; Fraga, L. M.; Woszezenki, C. R.; Oliveira, L.; Francisco, D. J.; Machado, G. J. C.; Nogueira, C. & da Glória Notargiacomo, M. (2002), 'Animated

- Pedagogical Agent in the Intelligent Virtual Teaching Environment', *Interactive Educational Multimedia* 4, 53-61.
- [8] Tolentino, L.; Birchfield, D.; Megowan-Romanowicz, C.; Johnson-Glenberg, M. C.; Kelliher, A. & Martinez, C. (2009), 'Teaching and Learning in the Mixed-Reality Science Classroom', *J Sci Educ Technol* 18, 501-517.
- [9] Fassbender, E.; Richards, D. & Kavakli, M. (2006), 'Game engineering approach to the effect of music on learning in virtual-immersive environments', *Proceedings of International Conference on Games Research and Development: CyberGames 2006*, 224-230.
- [10] Self, J. (1999), 'The Defining Characteristics of Intelligent Tutoring System Research: ITS care, precisely', *International Journal of Artificial Intelligence in Education* 10, 350-364.
- [11] Prevost, S.; Hodgson, P.; Cook, L. & Churchill, E. (1999), 'Face-to-Face Interfaces' *Computer Human Interaction - CHI*.
- [12] Paulus, T.; Horvitz, B. & Shi, M. (2006), 'Isn't it just like our situation?' Engagement and learning in an online story-based environment', *ETR&D* 54(4), 355-385.
- [13] Reategui, E.; Polonia, E. & Roland, L. (2007), 'The role of animated pedagogical agents in scenario-based language e-learning: a case-study', *ICL*.
- [14] Arroyo, I.; Woolf, B. P.; Royer, J. M. & Tai, M. (2009), 'Affective Gendered Learning Companions', *International Conference on Artificial Intelligence and Education, International Conference on Artificial Intelligence and Education*.

# Análisis de plagios en prácticas de programación

Manuel Rubio Sánchez

Departamento de Lenguajes y Sistemas Informáticos I  
Universidad Rey Juan Carlos  
Tulipán s/n, 28933 Madrid, España

**Resumen.** El copiar en instituciones académicas y en particular en asignaturas de programación es un fenómeno mundial. Además, es uno de los problemas más mencionados asociados a sistemas de evaluación automática de programas informáticos, que no suelen incorporar funcionalidad para detectar plagios. Sin embargo, los trabajos de la literatura que describen estas herramientas solo mencionan la necesidad de abordar la cuestión, sin profundizar en ella. En este artículo analizamos el alcance del problema en una asignatura sobre diseño y análisis de algoritmos en la que se ha empleado el juez automático Mooshak. Los resultados, basados en datos recopilados a lo largo de tres cursos académicos, muestran que el porcentaje de copias es alarmantemente elevado en grupos presenciales, aunque independientemente del uso de Mooshak. Aunque el problema es considerablemente menor en grupos a distancia, es fundamental no repetir ejercicios de años anteriores. El artículo plantea una estrategia sencilla pero eficaz que se apoya en herramientas de detección de plagios para prácticamente eliminar copias en grupos presenciales. Por último, se analiza la opinión de profesores y alumnos sobre la severidad de varios escenarios asociados a posibles copias y cómo penalizarlas.

## 1. Introducción

Las copias en entornos académicos constituyen un problema global que afecta a la efectividad de la enseñanza, genera evaluaciones injustas, o desmotiva a los alumnos que trabajan adecuadamente. En este artículo analizamos en detalle las copias que se han producido en una asignatura de diseño y análisis de algoritmos de la Universidad Rey Juan Carlos a lo largo de tres cursos académicos, y en el que se ha utilizado el sistema de evaluación automática Mooshak [8]. Estas herramientas (véase [5,1,7,2,6]) poseen varias ventajas tanto para estudiantes como para docentes, pero precisamente una de sus principales limitaciones está relacionada con las copias de códigos fuente, dado que no suelen incorporar funcionalidad para detectarlas. De esta manera, los trabajos de la literatura que describen estos sistemas generalmente solo mencionan la problemática sin profundizar en ella.

Se ha podido observar que el porcentaje de copias es llamativamente elevado, aunque similar al de otro grupo en el cual no se empleó Mooshak. Por este motivo en el tercer curso analizado se llevó a la práctica una estrategia sencilla pero eficaz que se apoya en las dos herramientas de detección de plagios de programas

informáticos más populares: Moss [10] y JPlaj [9], que prácticamente ha eliminado copias en grupos presenciales.

Por último, en asignaturas de programación el problema es complejo debido a que la colaboración entre alumnos suele ser beneficiosa, como han demostrado metodologías como la programación por parejas. En este sentido, no es trivial clasificar determinadas acciones como legales o penalizables. Por este motivo hemos analizado la opinión de profesores y alumnos sobre la severidad de varios escenarios asociados a posibles copias. Los profesores son en general más estrictos que los alumnos, por lo que es necesario especificar los criterios que se van a emplear en las guías docentes. Por otro lado, hemos observado una gran variabilidad en cuanto a cómo penalizar las copias por parte de los profesores, que puede dificultar la implantación de políticas de integridad académica.

## 2. Contexto

Eliminar las copias en cursos de programación es fundamental para que los alumnos asimilen conceptos y para que los más retrasados no se queden descolgados. En la Universidad Rey Juan Carlos llevamos tres años empleando el sistema de evaluación automática Mooshak, en algunos grupos de la asignatura “diseño y análisis de algoritmos”, el cual compila y ejecuta los programas de los alumnos frente a unos casos de prueba. Dado que los programas debían superar dichas pruebas obligatoriamente (para poder aprobar prácticas planteadas para realizarse de manera individual), muchos alumnos terminaban copiándolas. De esta manera, planteamos realizar un estudio para analizar cómo se producen las copias, y si el uso de Mooshak podría tener un efecto en ellas.

The screenshot shows the JPlag interface with the following components:

- Matches for:** Alumno36\_PracticaSkyline.java & Alumno11\_skyLine.java
- Similarity:** 55.7%
- Match Table:**

Alumno36_PracticaSkyline.java (60.0%)	Alumno11_skyLine.java (51.0%)	Tokens
Alumno36_PracticaSkyline.java(7-11)	Alumno11_skyLine.java(116-119)	10
Alumno36_PracticaSkyline.java(12-18)	Alumno11_skyLine.java(121-127)	18
Alumno36_PracticaSkyline.java(21-27)	Alumno11_skyLine.java(128-133)	10
Alumno36_PracticaSkyline.java(29-47)	Alumno11_skyLine.java(96-110)	20
- Code Views:** Two code editors side-by-side. The left editor shows the code for 'PracticaSkyline.java' and the right editor shows the code for 'skyLine.java'. Arrows point to matching lines between the two files.

Figura 1. Comparación de dos códigos con JPlag.

**Tabla 1.** Características de los grupos en los que se basa el estudio.

Grupo	Curso	Año	Nº de alumnos	Modalidad	Mooshak	Titulación	Campus
P11M	2010-11	2º	55	presencial	sí	GII	C1
P12M	2011-12	2º	44	presencial	sí	GIS	C1
P13	2012-13	2º	84	presencial	no	GII	C1
P13M	2012-13	2º	54	presencial	sí	GII	C2
O11M	2010-11	2º	19	a distancia	sí	GII	-
O12M	2011-12	2º	30	a distancia	sí	GII	-
O13M	2012-13	2º	27	a distancia	sí	GII	-
S13	2012-13	2º	13	presencial	no	GIS	C1
CS1	2012-13	1º	126	presencial	no	GII y GIC	C1

A pesar de los avisos (véase [12]) sobre la intención de penalizar las copias, detectamos un número elevado de ellas en los dos primeros años. De esta manera, empleamos una estrategia más agresiva para convencer a los alumnos de la intención de penalizarlas, en las que se llegó a mostrarles los resultados de JPlaj (véase la Fig. 1). En la interfaz se comparan dos códigos, donde fragmentos equivalentes se ilustran con el mismo color. Este enfoque ha conseguido reducir el número de copias considerablemente.

El elevado número de copias se puede deber a que los propios alumnos no sean conscientes de que están copiando [4]. Por tanto, se llevó a cabo una encuesta, tanto a alumnos como a profesores, para medir las diferencias a la hora de determinar la severidad de escenarios asociados a copias. Además, se midió cómo consideran que deberían penalizarse las copias y se preguntó a los profesores si han llegado a implementar dichas penalizaciones.

### 3. Descripción

El estudio se ha llevado a cabo en la Universidad Rey Juan Carlos durante los cursos académicos 2010-11, 2011-12, y 2012-13, con alumnos matriculados en la asignatura “diseño y análisis de algoritmos”, de segundo curso de grados en ingeniería informática (GII), y del software (GIS). La Tab. 1 muestra el número de alumnos matriculados, la modalidad de las clases, el grado, campus y curso en el que se impartieron las clases (las de GII se ofrecen en dos campus diferentes denotados C1 y C2), de los siete grupos en los que se analizaron las copias. Mientras que las poblaciones de alumnos para titulaciones presenciales y a distancia son claramente diferentes, suponemos que es la misma para todos los grupos presenciales. El grupo con el cual no se empleó Mooshak ha servido de control para medir el efecto del juez en las copias. La tabla incluye dos grupos adicionales de alumnos (S13 y CS1, este último con alumnos también de un grado en ingeniería de computadores (GIC)) que respondieron a un cuestionario sobre colaboración y copias (en estos casos se indica el número de alumnos que contestaron al cuestionario).

En la asignatura se planteaban una serie de prácticas obligatorias sobre diseño de algoritmos, generalmente con ejercicios de programación. Si se empleaba Mooshak las soluciones a los ejercicios debían pasar obligatoriamente las pruebas del

juez. En los dos primeros cursos los programas podían ser implementados en Pascal, C++ o Java, donde se asumió que no habría copias entre programas escritos en lenguajes diferentes. En el curso 2012-13 se restringió el lenguaje a Java.

Para determinar si dos códigos estaban involucrados en una copia en primer lugar se analizaron empleando Moss y JPlaj. Posteriormente, examinamos con cuidado cada pareja de códigos señalada como sospechosa, por un elevado índice de similitud, para tomar una decisión final. En este sentido, solo hemos considerado que dos códigos han sido copiados cuando hemos estado prácticamente seguros de que en efecto se ha producido la copia, dando a los alumnos el beneficio de la duda.

### 3.1. Análisis de copias

Dado que simples avisos o amenazas no suelen resultar fructíferas a la hora de reducir el número de copias [12], en el curso 2012-13 optamos por una estrategia más agresiva para transmitir al alumnado nuestra intención de detectar y penalizarlas. En concreto, se planteó una práctica opcional al principio del curso, con la cual los alumnos podían obtener un 5% adicional en la nota final. Tras realizarla se analizaron las copias con JPlag, y durante una sesión de laboratorio se fueron llamando a los alumnos involucrados en las copias halladas, para mostrarles los resultados de la propia interfaz de JPlag. A estos alumnos se les comentó que no recibirían puntos en la práctica, y se advirtió que si se volviese producir una copia suspenderían la asignatura. Este enfoque benigno para aquellos que copian la práctica opcional ha resultado efectivo a la hora de reducir el número de copias.

En cuanto a los grupos a distancia, en los dos primeros cursos apenas hubo copias. En el tercer curso se propusieron tanto ejercicios planteados en el curso anterior como nuevos, para medir una posible diferencia en la proporción de copias.

Los copias detectadas se han visualizado mediante gráficas donde se representan conjuntos de alumnos con códigos considerados como copiados. Por otro lado, hemos llevado a cabo tests chi-cuadrado de independencia (o tests exactos de Fisher en el caso trabajar con pocos datos) para medir diferencias en proporciones de copias.

### 3.2. Opinión de alumnos y profesores sobre colaboración y copias

La colaboración entre alumnos suele ser positiva, especialmente en informática, donde metodologías colaborativas como la programación por parejas han tenido éxito tanto en entornos académicos como profesionales. En este sentido, es normal que los alumnos intercambien información de alguna manera a la hora de realizar prácticas. No obstante, en ocasiones no es trivial definir la línea que separa una mera cooperación de una copia. Por este motivo analizamos la opinión de profesores y alumnos sobre el grado de severidad de diferentes escenarios relacionados con posibles copias a través de la encuesta mostrada en la Tab. 2. Las seis primeras preguntas se han tomado de [11]. Para comparar las respuestas entre grupos se han llevado a cabo tests U de Mann-Whitney.

**Tabla 2.** Encuesta sobre colaboración y copias.

Variable	Pregunta	Tipo	Sujetos
v1	Entregar una práctica previamente realizada en otra asignatura	L1	A y P
v2	Entregar una práctica previamente realizada por otro alumno en un curso anterior	L1	A y P
v3	Copiar material para una memoria de Internet	L1	A y P
v4	Utilizar una chuleta con datos importantes	L1	A y P
v5	Copiar material de un libro de texto para una memoria	L1	A y P
v6	Coordinar la realización de prácticas con un amigo, para que cada uno haga una por separado, en lugar de tener que hacer ambas	L1	A y P
v7	Copiar funciones auxiliares básicas o de entrada y salida de un amigo, pero implementar el algoritmo pedido yo mismo	L1	A y P
v8	Si un alumno copia una prueba solo debería suspender dicha prueba, y no la asignatura entera	L2	A y P
v9	Si un alumno copia una prueba durante la evaluación continua éste debería poder examinarse de la prueba en la segunda convocatoria	L2	A y P
v10	He aprobado a alumnos incluso sabiendo que han copiado prácticas	sí/no	P
v11	He aprobado a alumnos incluso sabiendo que han copiado durante un examen	sí/no	P

L1 = escala Likert (1: no copia, 5: copia).

L2 = escala Likert (1: muy en desacuerdo, 5: muy de acuerdo).

A = alumnos, P = profesores.

## 4. Resultados

A continuación se describen los resultados del estudio, así como la interpretación de los mismos.

### 4.1. Análisis de copias

La Fig. 2 muestra las copias detectadas en el grupo P11M, en siete ejercicios planteados a través de tres prácticas asociadas a temas de la asignatura: divide y vencerás (DV), programación dinámica (PD), y vuelta atrás o *backtracking* (VA). En particular, los ejercicios de DV consistieron en implementar (1) una potencia en tiempo logarítmico, (2) el método de bipartición para hallar ceros de una función, (3) multiplicación de matrices por bloques, y (4) una variante del algoritmo *mergesort*. Los ejercicios de PD pedían resolver el problema de la planificación de una cadena de montaje [3] siguiendo (1) una filosofía de arriba a abajo, y (2) de abajo a arriba. Por último, se pidió implementar un programa para resolver sudoku empleando la técnica de la vuelta atrás. La figura ilustra conjuntos de alumnos (cada uno representado mediante un determinado índice) cuyos códigos son tan semejantes que los hemos considerado como copiados. Para facilitar la localización de los alumnos en este tipo de gráficas cada uno solo aparece en una única fila, habiendo dos como mucho en una fila. Los alumnos unidos mediante guiones habían trabajado como pareja en una práctica previa, lo cual aporta mayor evidencia de copia.

En primer lugar, destaca el elevado porcentaje de códigos copiados, calculados sobre el número total de entregas en un ejercicio concreto, que da una idea sobre el alcance de la problemática. Teniendo en cuenta todos los ejercicios y todas las entregas el porcentaje asciende a  $96/211 = 45,5\%$ . Por otro lado, resulta difícil determinar posibles pautas o patrones a partir de los datos, dada la complejidad

conjuntos de códigos considerados como copiados	<div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">22</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">25</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">15</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">18</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">16</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">32</div>	<div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">22</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">25</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">15--21</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">18</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">16</div>	<div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">18</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">32</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">20</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">35</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">44</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">48--50</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">13</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">29</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">40</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">9</div>	<div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">18</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">32</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">20</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">44</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">48--50</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">13</div>	<div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">12</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">23</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">15</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">18</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">20--36</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">19--35</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">14--38</div>	<div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">15</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">18</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">20--36</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">19--35</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">14--38</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">4--37</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">39--24</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">47</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">1</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">28</div>	<div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">2--22</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">12</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">25</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">21</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">20--36</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">19--35</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">14--38</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">44</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">48--50</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">13</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">29</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">37</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">39--24</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">6--8</div>
	nº entregas	44	31	24	21	28	30
% copias	34,1%	48,4%	54,2%	33,3%	35,7%	50%	63,6%
<b>P11M</b>	<b>Ej 1</b>	<b>Ej 2</b>	<b>Ej 3</b>	<b>Ej 4</b>	<b>Ej 1</b>	<b>Ej 2</b>	<b>Ej 1</b>
		<b>DV</b>			<b>PD</b>		<b>VA</b>

**Figura 2.** Copias detectadas en el grupo P11M (curso 2010-11, presencial, con Mooshak). Se ilustran conjuntos de alumnos (representados mediante índices enteros) cuyos códigos son tan semejantes que los consideramos como copiados, para siete ejercicios englobados en tres prácticas: DV (divide y vencerás), PD (programación dinámica) y VA (vuelta atrás). Los alumnos unidos mediante guiones habían trabajado como pareja en una práctica previa.



conjuntos de códigos considerados como copiados				
	nº entregas	27	28	24
% copias	40%	7%	42%	54%
<b>P12M</b>	<b>Ej 1</b>	<b>Ej 2</b>	<b>Ej 1</b>	<b>Ej 1</b>
	<b>IM</b>		<b>DV</b>	<b>VA</b>

**Figura 3.** Copias detectadas en el grupo P12M (curso 2011-12, presencial, con Mooshak). Los ejercicios se engloban en tres prácticas: IM (introducción a Mooshak), DV (divide y vencerás) y VA (vuelta atrás).

de las redes formadas. En cualquier caso, se observa que los alumnos que copian suelen hacerlo en varios ejercicios. Los alumnos no solo copian por parejas, sino que pueden llegar a formar grupos numerosos (de hasta 7 alumnos en el cuarto ejercicio de la práctica de divide y vencerás). Estos grupos pueden surgir al interactuar en la propia aula, aunque también es posible que aparezcan debido a que el código copiado haya sido obtenido de una fuente común, por ejemplo, bajado de un sitio de Internet.

En el curso 2011-12 se decidió disminuir el número de ejercicios propuestos (y disminuir su dificultad ligeramente). Se plantearon dos ejercicios dentro de una práctica de introducción a Mooshak: (1) resolver el problema de las Torres de Hanoi mediante un algoritmo iterativo, y (2) cálculo de coeficientes binomiales. La práctica de divide y vencerás consistió en resolver el problema clásico “skyline”, mientras que la de backtracking pidió construir un verificador de sudokus (capaz de analizar el número de soluciones que puede llegar a tener un sudoku cuyo tablero inicial pudiera estar mal planteado).

La Fig. 3 muestra las copias detectadas en el grupo P12M. De nuevo destaca la variedad de agrupaciones, donde no se aprecian patrones claros. El porcentaje de envíos copiados por ejercicio ( $37/105 = 35,2\%$ ) decreció, pero no significativamente (con nivel de significación  $\alpha = 0,05$ ) con respecto al del curso anterior, como revela un test chi-cuadrado de independencia asociado a la tabla de contingencia en la Tab. 3, donde  $\chi^2 = 3,0277$ , y  $p = 0,082$ .

**Tabla 3.** Tabla de contingencia relacionando las copias y grupos con diferentes volúmenes de trabajo.

		Copia	No copia		
P11M (7 ejercicios)		96	115		
P12M (4 ejercicios)		37	68		
conjuntos de códigos considerados como copiados	(7 23)			(7 23)	
	(6 27)				
	(19 24)				
	(22 40)				
	(25 30)				
		(32)	(21)	(32 41)	
nº entregas	36	26	24	21	22
% copias	27,8%	3,9%	12,5%	0%	9,1%
<b>P13M</b>	<b>Ej 2</b> <b>IM (opcional)</b>	<b>Ej 1</b> <b>DV</b>	<b>Ej 1</b> <b>VA</b>	<b>Ej 1</b> <b>AV (nueva)</b>	<b>Ej 2</b>

**Figura 4.** Copias detectadas en el grupo P13M (curso 2012-13, presencial, con Mooshak). Los ejercicios son idénticos a los propuestos en el curso 2011-2012, aunque se añadieron dos sobre algoritmos voraces (AV). Los conjuntos de un elemento indican copias con respecto a algún código del curso anterior. El número de copias disminuye significativamente al detectar copias y avisar a los alumnos implicados.

En el curso 2012-13 se planteó la práctica opcional en la que se avisaba a los alumnos que copiaban, con el segundo ejercicio de la práctica de introducción a Mooshak propuesto el año anterior. Las prácticas sobre divide y vencerás y vuelta atrás también fueron las mismas que en el curso anterior. Además, se incorporó una práctica sobre algoritmos voraces con dos ejercicios. La Fig. 4 muestra los resultados. Hubo 10 copias en la práctica opcional, pero en el resto de ejercicios los porcentajes bajan considerablemente. Tanto el test chi-cuadrado ( $\chi^2 = 10,86$ , y  $p = 0,00098$ ) como el exacto de Fisher ( $p = 0,002$ ) indican que la diferencia es estadísticamente significativa. Los tests se basan en la tabla de contingencia mostrada en la Tab. 4. La diferencia también es significativa (con valores  $p$  menores),

**Tabla 4.** Tabla de contingencia para medir si emplear la estrategia asociada a la práctica opcional tiene un efecto sobre las copias en el grupo P13M.

	Copia	No copia
Práctica opcional y aviso	6	87
Sin práctica opcional ni aviso	10	26

conjuntos de códigos considerados como copiados	(25)	(13 19)
	(2 7)	(7)
	(5 36)	(5)
	(8 11)	
	(14 17)	(14 17)
	(29)	(25)
	(1)	(1 44)
	(9 12)	
	(32 34)	
	(6 39)	(22 30)
	(22)	
	(18 28)	
nº entregas	39	40
% copias	46%	33%
<b>P13</b>	<b>Ej 1 DV</b>	<b>Ej 1 VA</b>

**Figura 5.** Copias detectadas en el grupo P13 (curso 2012-13, presencial, sin Mooshak). Los ejercicios son idénticos a los propuestos en el curso 2011-2012. Conjuntos de un elemento indican copias con respecto a algún código del curso anterior.

**Tabla 5.** Tabla de contingencia para medir si Mooshak tiene un efecto sobre las copias en grupos presenciales.

	Copia	No copia
Con Mooshak	143	209
Sin Mooshak	31	48

si se incorporan datos de otros cursos o del grupo en el que no se empleó Mooshak (en los que no se avisó).

La Fig. 5 muestra las copias detectadas en el grupo presencial que no usó Mooshak. Los porcentajes de copias son similares a los de otros grupos en los que sí se utilizó el juez. El test chi-cuadrado basado en la tabla de contingencia de la Tab. 5, en la que se han usado datos de los grupos P11M, P12M, y del ejercicio opcional en el grupo P13M, indica que usar Mooshak no tiene un efecto sobre las copias ( $\chi^2 = 0,05$ , y  $p = 0,82$ ).

en cuanto a los grupos a distancia, no detectamos ninguna copia en el curso 2010-11. Esto se debe tanto al pequeño número de envíos a Mooshak (20 repartidos en 7 ejercicios), como al hecho de que no se plantearon ejercicios propuestos en años anteriores. El siguiente curso solo copiaron dos alumnos en tres ejercicios, tal y como muestra la Fig. 6. Los ejercicios fueron idénticos a los propuestos en el grupo presencial, y diferentes a los pedidos el año anterior.

conjuntos de códigos considerados como copiados	16 18	16 18		16 18
nº entregas	13	10	11	10
% copias	15%	20%	0%	20%
<b>O12M</b>	<b>Ej 1</b>	<b>Ej 2</b>	<b>Ej 1</b>	<b>Ej 1</b>
	<b>IM</b>		<b>DV</b>	<b>VA</b>

**Figura 6.** Copias detectadas en el grupo O12M (curso 2011-12, a distancia, con Mooshak). Los ejercicios son idénticos a los propuestos en el grupo presencial.

**Tabla 6.** Tabla de contingencia para medir si los ejercicios propuestos otros años se copian más en el grupo O13M.

	Copia	No copia
Ejercicios repetidos	25	28
Ejercicios nuevos	4	24

Sin embargo, al proponer los mismos ejercicios en el curso 2012-13 que en el 2011-12 el número de copias aumentó considerablemente, como ilustra la Fig. 7. Para comparar los últimos dos ejercicios novedosos con el resto (propuestos el año anterior) se emplea la tabla de contingencia de la Tab. 6. El test exacto de Fisher indica un valor  $p$  igual a 0.0036. Por tanto, el número de copias claramente desciende si los ejercicios son nuevos para los alumnos. En caso de incluir datos de otros años, en los que los ejercicios eran novedosos y apenas se produjeron copias, el resultado de los tests también es significativo, con valores  $p$  más pequeños.

Por otro lado, hemos analizado las copias dividiendo a los alumnos en función de si se han matriculado por primera vez en la asignatura. Como los ejercicios propuestos en años anteriores serían novedosos para estos alumnos, cabe esperar que copien menos. El test exacto de Fisher indica que así es, a partir de la tabla de contingencia en la Tab. 7, ya que la diferencia entre las copias para estos alumnos es significativa ( $p = 0,002$ ). En cambio, la diferencia no resultó significativa al dividir a los alumnos en función de si habían agotado alguna convocatoria.

**Tabla 7.** Tabla de contingencia para medir si los alumnos de primera matrícula copian menos que el resto en ejercicios propuestos en años anteriores.

	Copia	No copia
Primera matrícula	4	17
Segunda matrícula o más	21	11

conjuntos de códigos considerados como copiados	$\begin{matrix} 1 & 6 \\ 13 & 19 \end{matrix}$	$\begin{matrix} 19 \\ 4 & 18 \end{matrix}$	$\begin{matrix} 4 & 18 \\ 10 & 17 \\ 7 \\ 2 & 5 \end{matrix}$	$\begin{matrix} 4 & 18 \\ 4 & 18 \end{matrix}$	
	$\begin{matrix} 4 & 18 \\ 16 \\ 10 & 17 \end{matrix}$	$\begin{matrix} 17 \\ 12 \\ 2 & 5 \\ 8 & 9 \end{matrix}$			
nº entregas	17	18	18	13	15
% copias	53%	50%	39%	15%	13%
<b>O13M</b>	<b>Ej 2 IM</b>	<b>Ej 1 DV</b>	<b>Ej 1 VA</b>	<b>Ej 1 AV (nueva)</b>	<b>Ej 2</b>

**Figura 7.** Copias detectadas en el grupo O13M (curso 2012-13, a distancia, con Mooshak). Los ejercicios son idénticos a los propuestos en el grupo presencial. La práctica sobre algoritmos voraces es la única que contiene ejercicios diferentes a los planteados en el curso anterior.

**Tabla 8.** Porcentajes de copia en los ejercicios planteado en los siete grupos analizados.

Práctica	IM		DV				PD		VA	AV		Total
Ejercicio	1	2	1	2	3	4	1	2	1	1	2	
P11M	-	-	34.1	48.4	54.2	33.3	35.7	50	63.6	-	-	45.5
P12M	40.7	7.1	41.7	-	-	-	-	-	54	-	-	35.2
P13	-	-	46.2	-	-	-	-	-	32.5	-	-	39.2
P13M	-	27.8	3.9	-	-	-	-	-	12.5	0	9.1	27,8(IM), 6,5(resto)
O11M	-	-	0	0	0	0	0	0	0	-	-	0
O12M	15.4	20	0	-	-	-	-	-	20	-	-	13.6
O13M	-	52.9	50	-	-	-	-	-	38.9	15.4	13.3	47.2(resto), 14.3(AV)

Finalmente, los porcentajes de copias quedan resumidos en la Tab. 8. Los guiones indican que no se planteó o no se analizaron las copias para un ejercicio.

#### 4.2. Opinión de alumnos y profesores sobre colaboración y copias

Es de esperar que los profesores sean más estrictos que los alumnos a la hora de valorar la severidad de los escenarios de posibles copias. En efecto, la mediana para los profesores es mayor o igual que la de los alumnos (de segundo) en las siete primeras variables de la Tab. 2. La Tab. 9 muestra el resultado de aplicar tests U de Mann-Whitney para determinar si hay diferencias significativas entre las respuestas (aplicando la corrección de Holm todas las diferencias siguen siendo significativas). En este estudio se distingue entre los alumnos a los que se aplicó la estrategia de proponer una práctica opcional y avisar de la intención de revisar

**Tabla 9.** Diferencias significativas entre opiniones de alumnos y profesores. Se muestran los valores  $p$  y las medianas de los grupos.

Variable	Valor $p$ del test U	(23) Profesores	(66) Alumnos sin practica opcional ni aviso
v2	0.000852	5	5
v3	0.000609	3	4
v5	0.000248	2	3
v6	0.005554	3	4
v7	0.010848	2	3

Variable	Valor $p$ del test U	(23) Profesores	(30) Alumnos con práctica opcional y aviso
v2	0.005502	5	5

**Tabla 10.** Diferencias significativas entre alumnos (de P13) que copian y los que no.

Variable	Valor $p$ del test U	Mediana de (19) alumnos que copiaron	Mediana de (17) alumnos que no copiaron
v2	0.005736	4	5
v4	0.001357	4	5

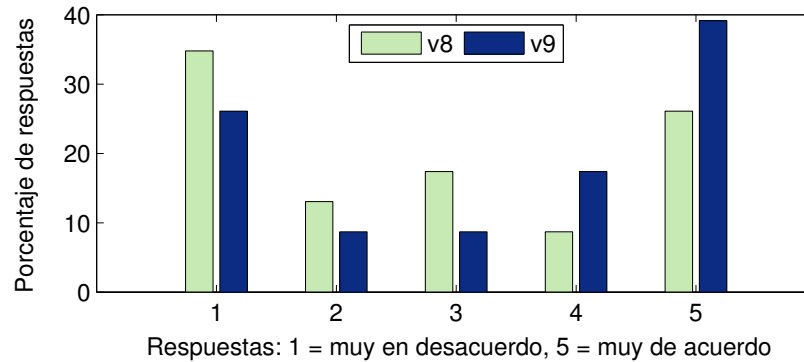
cuidadosamente las copias (30 pertenecientes al grupo P13M), y a los que no (53 pertenecientes al grupo P13 y otros 13 del grupo S13). Destaca que hay muchas más diferencias significativas en el grupo al que no se avisó. Por tanto, es posible que la estrategia (que dio frutos en cuanto a la reducción de copias) haya tenido un efecto sobre la forma de responder al cuestionario o incluso de entender la problemática por parte de los alumnos, donde su opinión se acerca más a la de los profesores.

También existen diferencias significativas entre las respuestas de alumnos que copiaron frente a los que no. Éstas se muestran en la Tab. 10, donde solo se han comparado las respuestas del grupo P13. Esto da un indicio sobre la posibilidad de predecir si un alumno va a copiar aprovechando información de este tipo de cuestionarios. No obstante, este análisis queda fuera del alcance de este artículo.

La Tab. 11 muestra diferencias interesantes asociadas a las variables v8 y v9 sobre los castigos a aplicar en caso de detectar una copia. En este caso se emplearon encuestas realizadas por alumnos de primer curso (al no disponer de dicha información para los grupos de segundo analizados en el resto del artículo). Es lógico

**Tabla 11.** Diferencias de opiniones sobre los castigos a tomar en caso de detectar copias.

Variable	Valor $p$ del test U	Mediana de (23) profesores	Mediana de (126) alumnos de CS1
v8	0.000001	3	5
v9	0.000893	4	5



**Figura 8.** Histogramas de las respuestas de los profesores a las preguntas v8 y v9.

que los profesores sean más exigentes. No obstante, la opinión de los profesores varía considerablemente. La Fig. 8 muestra un histograma con los porcentajes de respuestas a las preguntas v8 y v9, donde se puede apreciar como su forma es claramente multimodal, con más masa de probabilidad en los extremos. Esto señala la dificultad que supondría llegar a acuerdos para regular la actuación ante las copias u otras actividades de fraude académico dentro de departamentos, escuelas o facultades.

Finalmente, algunos profesores no toman medidas en caso de detectar copias, lo cual puede incitar a los alumnos a incurrir en ellas. Siete de los 23 profesores admite haber aprobado a alumnos sabiendo que han copiado prácticas (v10), e incluso dos de ellos lo han hecho habiendo detectado copias en exámenes (v11).

## 5. Conclusiones

El estudio ha mostrado que los sistemas de evaluación automática no causan un incremento en el porcentaje de copias, incluso cuando se exige que los programas superen pruebas de caja negra. Por otro lado, para reducir las copias ha resultado efectiva una estrategia fundamentada en mostrar a los alumnos una intención seria y creíble a la hora de penalizarlas.

Proponer ejercicios originales y novedosos para los alumnos es una recomendación básica, que ha quedado de manifiesto en este estudio. En los cursos presenciales este efecto apenas se ha notado debido a que las clases han sido en campus o titulaciones diferentes de un año al siguiente. De no haberse dado esta circunstancia el problema podría haber sido mayor, a tenor del porcentaje de copias que se producen incluso cuando los ejercicios son novedosos.

Finalmente, la opinión de los profesores varía considerablemente a la hora de penalizar las copias. Como sería difícil llegar a posturas comunes en cuanto a la toma de medidas, debe ser el propio profesor quien especifique en la guía docente de la asignatura las actuaciones que considere oportunas. Pensamos que incluso sería

aconsejable incluir una lista (no cerrada) de actividades no permitidas. Aunque en el sistema universitario público español existe un apartado para especificar el método de evaluación en las guías docentes, no hay ninguno explícito para indicar las acciones a tomar en casos de fraude académico (lo cual es muy común en otros países, como por ejemplo, en Estados Unidos). Pensamos que la incorporación de este apartado sería beneficiosa para reducir la problemática asociada a las copias.

## Agradecimientos

Este trabajo ha sido financiado con el proyecto TIN2011-29542-C02-01.

## Referencias

1. Kirsti M. Ala-Mutka. A survey of automatic assessment approaches for programming assignments. *Computer Science Education*, 15(2):83–102, 2005.
2. Peter Brusilovsky and Colin Higgins. Preface to the special issue on automated assessment of programming assignments. *Journal of Educational Resources in Computing*, 5(3), September 2005. Article 1.
3. Thomas H. Cormen, Clifford Stein, Ronald L. Rivest, and Charles E. Leiserson. *Introduction to Algorithms*. The MIT Press, 2nd edition, 2001.
4. Maureen M. Dawson and Joyce A. Overfield. Plagiarism: Do students know what it is? *Bioscience Education*, 8, 2006.
5. Christopher Douce, David Livingstone, and James Orwell. Automatic test-based assessment of programming: A review. *Journal of Educational Resources in Computing*, 5(3), September 2005. Article 4.
6. Petri Ihantola, Tuukka Ahoniemi, Ville Karavirta, and Otto Seppälä. Review of recent systems for automatic assessment of programming assignments. In *Proceedings of the 10th Koli Calling International Conference on Computing Education Research*, Koli Calling '10, pages 86–93, New York, NY, USA, 2010. ACM.
7. Mike Joy, Nathan Griffiths, and Russell Boyatt. The boss online submission and assessment system. *Journal of Educational Resources in Computing*, 5(3), September 2005. Article 2.
8. José Paulo Leal and Fernando Silva. Mooshak: a web-based multi-site programming contest system. *Software-Practice & Experience*, 33:567–581, May 2003.
9. Lutz Prechelt, Guido Malpohl, and Michael Philippsen. Finding plagiarisms among a set of programs with jplag. *Journal of Universal Computer Science*, 8:1016–1038, 2000.
10. Saul Schleimer, Daniel S. Wilkerson, and Alex Aiken. Winnowing: Local algorithms for document fingerprinting. In *Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data 2003*, pages 76–85. ACM Press, 2003.
11. Judy Sheard, Martin Dick, Selby Markham, Ian Macdonald, and Meaghan Walsh. Cheating and plagiarism: perceptions and practices of first year it students. *SIGCSE Bulletin*, 34:183–187, June 2002.
12. Robert J. Youmans. Does the adoption of plagiarism-detection software in higher education reduce plagiarism? *Studies in Higher Education*, 36(7):749–761, 2011.