

CB 28**ESTADO DEL ARTE DE APLICACIONES PARA EL APRENDIZAJE DE CÁLCULO NUMÉRICO. ANÁLISIS DE UN CASO PARTICULAR**

**María Eva ASCHERI, Rubén PIZARRO, Gustavo ASTUDILLO, Pablo GARCÍA,
María E. CULLA**

Facultad de Ciencias Exactas y Naturales – Universidad Nacional de La Pampa
Uruguay 151 – Santa Rosa La Pampa – TE: (02954) 425166 Int. 125
mavacheri@exactas.unlpam.edu.ar ruben@exactas.unlpam.edu.ar
astudillo@exactas.unlpam.edu.ar pablogarcia@exactas.unlpam.edu.ar
cullam@exactas.unlpam.edu.ar

Palabras Clave: software educativo, herramientas gratuitas, ecuaciones no lineales, interpolación polinómica, ajuste de curvas.

RESUMEN

En el marco de un Proyecto de Investigación que desarrollamos en la Facultad de Ciencias Exactas y Naturales de la UNLPam, comenzamos a diseñar y elaborar un software educativo utilizando herramientas computacionales gratuitas. En dicho software, implementamos los métodos numéricos clásicos para la resolución de ecuaciones no lineales, interpolación y aproximación polinomial y ajuste de curvas. Lo utilizamos en el desarrollo de Cálculo Numérico que se dicta para estudiantes de Profesorado en Matemática, Licenciatura en Física e Ingeniería Civil. A través de su uso, nos propusimos alcanzar los siguientes objetivos:

1. Generar el contexto educativo acorde a los contenidos a desarrollar y a los objetivos propuestos en la asignatura.
2. Proporcionar una herramienta tecnológica para afianzar los conceptos teóricos y las técnicas utilizadas en la aplicación de los diferentes métodos numéricos.
3. Facilitar la enseñanza y el aprendizaje de los diversos métodos numéricos por medio de la visualización gráfica del funcionamiento de los mismos.

En este trabajo, presentamos las principales características del software educativo de elaboración propia y su uso, la metodología empleada para su diseño, el estado del arte de aplicaciones para el aprendizaje de Cálculo Numérico y de este software en particular, y consideraciones finales.

INTRODUCCIÓN

Comenzamos con la elaboración del software educativo teniendo como objetivo disponer de un material didáctico de acceso gratuito desarrollado, específicamente, para facilitar el proceso de enseñanza y aprendizaje de los temas de Cálculo Numérico: *Resolución Numérica de Ecuaciones no Lineales, Interpolación y Aproximación Polinomial y Ajuste de Curvas por Mínimos Cuadrados*. Esta aplicación debía poder usarse libremente como herramienta para la enseñanza y el aprendizaje de algunos métodos numéricos, en donde se mostrase de forma numérica y gráfica el comportamiento de los mismos.

Previo al desarrollo del primer prototipo del software, se realizó una búsqueda exhaustiva a través de la Web con el objetivo de hacer un relevamiento de las aplicaciones disponibles para trabajar temas de Cálculo Numérico (Ascheri et al., 2007). Los resultados obtenidos, nos

llevaron a la decisión de implementar una aplicación Web utilizando HTML, PHP y la librería JPGRAPH¹.

Una vez concretado el primer prototipo –opciones para el cálculo de raíces de ecuaciones no lineales– (Ascheri et al., 2008a y b), se realizó un proceso de evaluación que nos permitió detectar debilidades y fortalezas de la aplicación (Ascheri et al., 2009 y 2010), y generar un segundo prototipo –se le agrega la opción de interpolación polinomial.

Partiendo de la evaluación mencionada en el párrafo anterior y a partir de la aplicación del software en el aula (Ascheri et al., 2011), pudimos evolucionar hasta completar la aplicación –se le incorpora la opción de ajuste de curvas por mínimos cuadrados.

La aplicación se encuentra disponible en el campus virtual de la Facultad de Ciencias Exactas y Naturales de la UNLPam en <http://online2.exactas.unlpam.edu.ar/numerico/>. Este software educativo se utiliza, fundamentalmente, durante el desarrollo del curso de Cálculo Numérico que se dicta para las carreras de Profesorado en Matemática (3° Año), Licenciatura en Física (3° Año) e Ingeniería Civil (2° Año) de la mencionada Facultad. Puesto que está disponible en la Web, también lo pueden usar otras disciplinas científicas.

MARCO TEÓRICO

Para definir y clasificar a los software educativos, Squires y Mc Dougall (1994), citados por Bosco (2002) y Cataldi (2000), hacen una revisión de un conjunto de marcos de referencia que agrupan en clasificación por tipo de aplicación, por su función educativa y por su fundamentación educativa. En la primera clasificación, se distinguen dos tipos de software educativos:

- El carente de contenidos, que es aquél que sin ser diseñado con una intencionalidad pedagógica es adaptado por el docente para sus clases, el cual no es considerado aquí como software educativo.
- El software específico para una asignatura, que es aquél que es diseñado específicamente para la enseñanza y el aprendizaje de un tema concreto de una asignatura. No hay acuerdo entre los autores en una taxonomía exacta, pero podríamos dar la siguiente clasificación:
 - ✓ **Ejercitación.** Aplicaciones que permiten la ejercitación y/o la resolución de problemas a través de una batería de enunciados con los que cuentan (en ocasiones, se pueden modificar). Ofrecen una respuesta inmediata al alumno/a en términos de correcto o incorrecto. Ejemplo de ellos son programas para practicar operaciones matemáticas o las estructuras gramaticales de un lenguaje.
 - ✓ **Tutoriales.** Son programas que guían al alumno/a en el desarrollo de uno o varios temas y pueden ofrecer o no un espacio para la ejercitación. Un ejemplo de este tipo de aplicaciones son las presentaciones multimedia diseñadas para el auto aprendizaje de algún tema o las aplicaciones del tipo “visita guiada”.
 - ✓ **Bases de datos.** Estos programas proporcionan datos organizados que permiten realizar consultas sobre ellos a través de palabras claves. Aquí se encuentran los programas como las enciclopedias.
 - ✓ **Simulaciones.** Los simuladores intentan representar lo más fielmente posible un aspecto de una construcción humana. La construcción simulada debe ser costosa o peligrosa de crear o manipular en la vida real y debe permitir el cambio de parámetros que definen la construcción. Ejemplo de esto son las aplicaciones que permiten entender el comportamiento de estructuras bajo

¹ Disponible en: <http://jpgraph.net/>

diferentes situaciones, juegos de video que simulan una sociedad o laboratorios virtuales.

- ✓ **Juegos.** Si bien son aplicaciones destinadas al divertimento, como sabemos a través de lo lúdico también se aprende. Además, existen juegos de video destinados específicamente al aprendizaje de algún tema a través del juego.
- ✓ **Inteligencia artificial.** Estas aplicaciones intentan reproducir el razonamiento del experto a través de la utilización de diferentes técnicas de la inteligencia artificial. Ejemplo de esto son los demostradores automáticos de teoremas.

Es importante recordar que algunos de los ejemplos que hemos expuesto en los ítems anteriores pueden no corresponderse con la definición dada para software educativo, pero fueron así considerados por los creadores del conjunto de marcos de referencia.

Si bien este enfoque brinda una manera sencilla de clasificar y describir el software educativo, también enfrenta algunos problemas. No cuenta con fundamentos claros, es muy sensible al paso del tiempo y, por ende, las nuevas aplicaciones no encajan en las categorías propuestas (Squires y Mc Dougall, 1994).

La clasificación por su función educativa, hace hincapié en la función del software y no se tiene en cuenta las necesidades de los alumnos/as ni las del docente, el diseñador crea la aplicación en función de lo que él cree que debería hacer el software. Este enfoque se vuelve riesgoso cuando es utilizado para el diseño de software educativo ya que rompe con un principio básico del propio diseño de software: tener en cuenta al usuario.

Por su parte, Sancho (1992), citado por Rodino (1996), divide también, en tres grupos las funciones que pueden desempeñar las computadoras (y el software en particular) en la enseñanza:

- Aplicaciones destinadas a la ejercitación, que tienen la función de refuerzo, control y prueba, donde el alumno/a realiza actividades que le permiten repasar, recordar y practicar. El tipo de aprendizaje propuesto por estas aplicaciones es estímulo-respuesta por repetición.
- Aplicaciones que permiten la verificación de hipótesis y la resolución de problemas, donde el alumno/a realiza actividades que le permiten aplicar el conocimiento y comprender como éste es aplicado para la resolución de problemas. El tipo de aprendizaje es por descubrimiento y resolución de problemas.
- Aplicaciones que permiten el análisis de problemas, donde el alumno/a realiza tareas en las que puede aplicar su conociendo. El tipo de aprendizaje que estimulan estas aplicaciones es el procesamiento significativo de la información.

Independientemente de las habilidades cognitivas que estimulen las actividades propuestas en la aplicación, siempre es posible que los alumnos/as cometan errores. Estos pueden tener que ver con la operación de la aplicación o con el cómo aplicar su conocimiento. Es por esto que, un aspecto muy importante a tener en cuenta al diseñar un software es el tratamiento que se le dará al error. Pero debemos aclarar que el tratamiento que se le de al error en la aplicación va a depender del marco de referencia que utilice el diseñador. Si el diseñador se encuentra dentro de un marco de referencia *conductista*, el error será penalizado pues no forma parte del aprendizaje; por el contrario, otro será el tratamiento del mismo para las aplicaciones enmarcadas dentro del *constructivismo* o el *cognitivismo* donde el error forma parte del proceso de aprendizaje. En este trabajo, consideramos que el error ocupa un lugar muy importante en el proceso de aprendizaje.

Si bien el manejo de errores tiene un extenso tratamiento dentro del diseño de software, creemos que cobra mayor importancia al ser un software educativo, ya que se tratan de errores cometidos por alumnos/as en su proceso de aprendizaje. En este sentido, Sternberg (1990), citado por Camilloni (1995), dice que los errores son ventanas que permiten observar el proceso a través del cual se realiza una tarea. De allí la importancia de tenerlos en cuenta al

momento de diseñar una aplicación educativa. Si bien los errores al operar un software pueden devenir de la falta de conocimiento o experiencia en el manejo del mismo, y éstos deben ser previstos y tratados por el programador, los que nos interesan aquí son aquellos que tienen que ver con el proceso de aprendizaje. Según Camilloni (1995), podemos encontrarnos con errores *inteligentes* y *no inteligentes*. En el primer caso, se comete el error porque se sabe algo. Es el caso donde un algoritmo (o un heurístico) que se conoce y/o que se ha aplicado con éxito en algún problema, es mal aplicado en la resolución de otro problema. En el segundo caso, los errores no inteligentes, son los que se producen por distracción o por azar. Esta autora dice que *“para estos errores es muy poco lo que podemos hacer”*. Sin embargo, creemos que el diseñador de un software educativo debe poner empeño en prever los errores y tratar de informarlos claramente al alumno/a de manera que éste pueda volver sobre sus pasos y seguir utilizando la aplicación.

Marquès (1996), nos plantea una interesante clasificación de software educativo en función del tratamiento del error. Habla de *programas directivos*, que controlan en todo momento la actividad del alumno/a y *programas no directivos*, donde la aplicación se concentra en procesar los datos y mostrar los resultados, no en *“juzgar las acciones del alumno”*. Si bien Marquès no especifica cómo es el tratamiento del error, nos resulta interesante este último planteo donde la aplicación no cumple la función de juez sino un facilitador de los aprendizajes.

Desde la definición dada por Marquès hasta esta última clasificación, propuesta también por él, hemos hablado de software educativo en general. Pero existe una ciencia a la que el software la ha acompañado desde sus inicios, la matemática. Como escribe Villegas (2007, p.149), *“para trabajar las matemáticas, el ser humano siempre ha necesitado de herramientas que le faciliten la representación y comprensión de los conceptos o le permitan aumentar la rapidez de cálculo”*. Esto se aplica directamente a las computadoras, que ya atraviesan todos los aspectos de la vida diaria, y lo hacen particularmente en la matemática. La computadora nace como herramienta de cálculo para posteriormente evolucionar hasta lo que conocemos hoy en día. Y, en esa evolución, siempre necesitó a la matemática y ésta a la computadora. Desde la criptografía, pasando por las hojas de cálculo, hasta llegar a demostradores de teoremas. Al estar tan ligadas la matemática y las computadoras, estas últimas no podían estar ajenas a los procesos de enseñanza y aprendizaje de la matemática. Dice Santandreu (2004, p.65), *“el uso de los recursos tecnológicos favorece un aprendizaje más eficiente en los alumnos e influyen sobre qué matemáticas enseñar y cómo hacerlo, son instrumentos de mediación en la construcción y estructuración del conocimiento matemático, suministran un nuevo ambiente de aprendizaje”*. En este sentido, Quintana Mendoza y Mendoza Bolo (2006, p.41) afirman que *“con apoyo de las computadoras se pueden adquirir conocimientos matemáticos que con la pizarra, tizas, papel o lápices sería muy limitado y poco agradable a la vista de los estudiantes”*.

No podemos dejar de mencionar que dentro del software para matemática se puede plantear una clasificación o una división en diferentes categorías. Esta clasificación surge en virtud de que el software para matemática se torna cada vez más específico. Varios autores (Losada Liste, 2007; Zhao, 1998; Balderas Puga, 1999) plantean clasificarlos de acuerdo a la temática o área de conocimiento (de la matemática) que abordan. Ellos clasifican al software para matemática en:

- CAS (Computer Algebra System/Sistemas de Álgebra Computacional)
- NCS (Numerical computation systems/Sistema para cálculo numérico)
- DGS (Dynamic Geometry System/Sistemas de Geometría Dinámica)
- APS (Automatic or Assistant proving systems/Demostradores automáticos de teoremas)
- GDAS (Graphics and data analysis systems/Paquetes estadísticos)
- SC (Symbolic calculators/Sistemas de cálculo simbólico)

- MES (Mathematics Education Systems/Software educativo para Matemática).

Esta clasificación, como ya se mencionó antes, corre el riesgo de volverse obsoleta y dejar afuera de la clasificación a nuevos software o bien aparecen aplicaciones que pueden ser categorizadas en más de un clasificador. Además, en nuestro caso, la clasificación para software educativo (MES) es un tanto general, y creemos que debiera ser asumida como transversal a las demás.

En el contexto de la clasificación de software, se puede hablar también de *software libre* y de *software propietario*. En la primera categoría, entran aquellos programas que permiten al usuario ejecutar, estudiar, redistribuir y mejorar el software. Y en la segunda categoría, se encuentran aquellos para los cuales debe comprarse una licencia para su utilización (Gómez Sánchez, 2004).

El software educativo de diseño propio. Principales características y su uso

En la Figura 1, vemos el primer menú que nos aparece al iniciar la navegación por el sitio:



Figura 1

Ingresando a la opción “Cálculo de raíces” aparece la ventana que nos permite elegir alguno de los seis métodos que implementamos para la resolución de ecuaciones no lineales: **bisección, iterativo de punto fijo, secante, regula-falsi, Newton-Raphson y de von Mises**. Al seleccionar, por ejemplo, el método de “Newton-Raphson” aparece una nueva ventana, como se muestra en la Figura 2, en la cual se ingresan los datos necesarios para poder implementarlo.



Figura 2

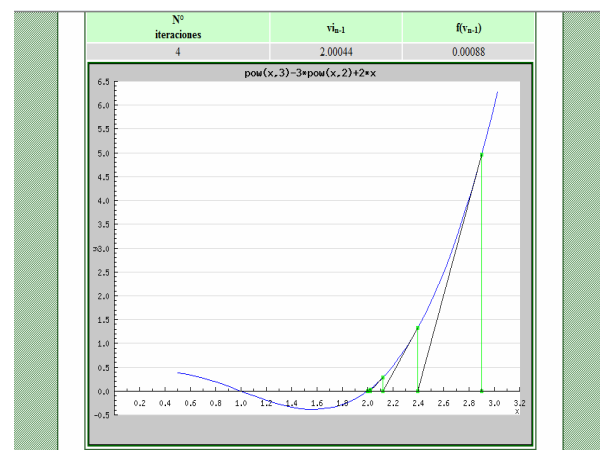


Figura 3

Al seleccionar la opción “**Aplicar el método**”, se obtienen sucesivas imágenes en las cuales se representa la gráfica de la función y las correspondientes aproximaciones a la raíz que se está buscando, obteniendo en este caso el gráfico de la Figura 3.

Finalmente, se muestran los datos numéricos relacionados con la resolución del problema propuesto, como se muestra en la Figura 4:



Figura 4

Si elegimos alguno de los otros métodos, el software responde similarmente según el problema a resolver.

Si en la Figura 1 seleccionamos la opción “**Interpolación**”, aparece una nueva ventana que nos permite elegir uno de los métodos de interpolación incluidos en el software, tales como el **método de interpolación de Newton** y el **método de interpolación de Lagrange**. Si elegimos este último, obtenemos la siguiente pantalla (Figura 5):

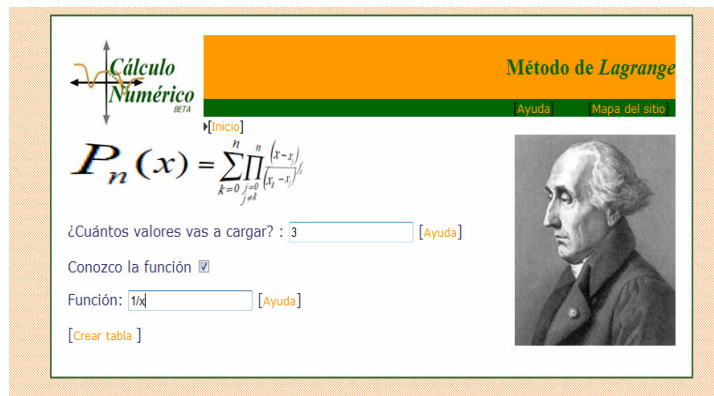


Figura 5

Podemos elegir la opción de interpolar aplicando la fórmula de Lagrange, conociendo o no la función. De esta forma, si seleccionamos la opción en la que conocemos la función e ingresamos la misma, los valores en la cual la queremos calcular y el valor a interpolar, obtenemos los resultados que se muestran en las Figuras 6 y 7:



Figura 6

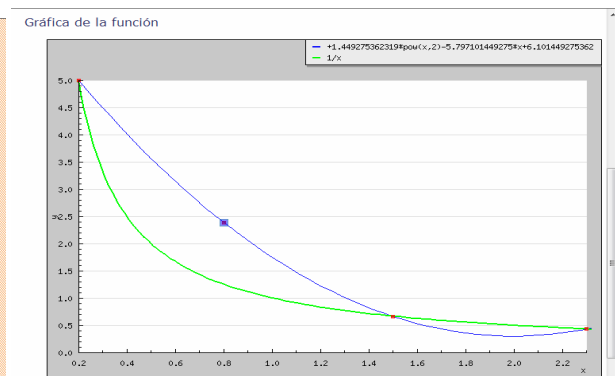


Figura 7

En la Figura 6, vemos los resultados numéricos que arroja el software, incluyendo el polinomio de interpolación obtenido. En la Figura 7, observamos la representación gráfica de la función conocida, del polinomio obtenido y del resultado del valor interpolado.

Por último, para realizar el ajuste de curvas por el método de mínimos cuadrados accedemos a la opción “**Ajuste de curvas**” (ver Figura 1). Luego de cargar la tabla de valores, el software grafica los puntos ingresados para que el usuario seleccione el tipo de ajuste que realizará (polinómico o exponencial), como se muestra en la Figura 8.



Figura 8

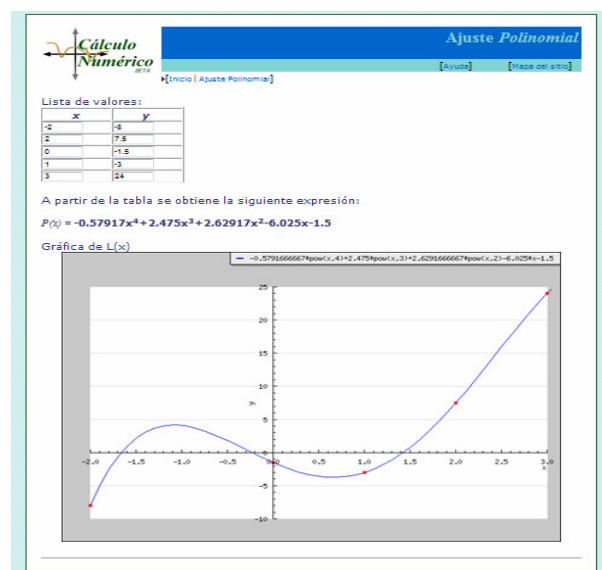


Figura 9

En la Figura 9 se muestra el resultado luego de elegir, en este caso, el ajuste polinómico.

En la parte superior derecha de las distintas pantallas mostradas (Figuras 1, 2, 5, 6, 8 y 9), se observa que se puede elegir la opción **Ayuda**. En ella se encuentra una descripción de lo que nos permite realizar el software, una síntesis teórica del método que se va a aplicar y de cómo utilizar el software para cada caso en particular.

Las actividades desarrolladas en Cálculo Numérico requieren de la realización de muchos y repetitivos cálculos. Se estudian diferentes métodos numéricos para obtener soluciones aproximadas a problemas que analíticamente no pueden resolverse o que resulta muy complejo poder hacerlo. Previo a la implementación del software educativo, durante el cursado de esta materia, los alumnos/as estudiaban los distintos métodos numéricos aplicando las diferentes fórmulas y elaboraban programas sencillos que resolvían numéricamente los problemas planteados. En todos los casos, advertimos, desde la cátedra, que la búsqueda de la solución esperada hacía que se perdiera de vista el funcionamiento de los métodos, su interpretación gráfica, las condiciones para su aplicación, el error cometido, entre otros. Esto nos motivó a que nos planteáramos diseñar un software educativo acorde a las necesidades y objetivos propuestos para el desarrollo de Cálculo Numérico. Por ello, con este software educativo, esperamos lograr un estudio minucioso de los aspectos gráficos de los diferentes métodos analizados.

Como vimos en las figuras anteriores es necesario considerar una serie de requerimientos para aplicar los métodos numéricos. Una vez que el usuario determine estos requerimientos, verá si la respuesta obtenida es adecuada o no, de acuerdo al problema que intenta resolver. Pretendemos así, que el alumno/a realice una revisión de los conceptos teóricos para confirmar por qué un método está funcionando o no, cuál de los métodos resulta más adecuado para resolver el problema en cuestión y hacer un análisis gráfico de cómo se obtienen los resultados.

Este software es utilizado en las clases teóricas y prácticas. En estas últimas, se resuelven ejercicios de los trabajos prácticos y de las evaluaciones parciales, mientras que para el desarrollo de la teoría, se utiliza el software para la interpretación geométrica de los métodos numéricos estudiados, para el planteo y el desarrollo de nuevos ejemplos.

METODOLOGÍA

Para la revisión de aplicaciones de éste trabajo, se utilizaron dos extensos compilados de aplicaciones que ofrece la enciclopedia virtual Wikipedia: *Comparison of numerical analysis software* (Wikipedia, 2012a) y *List of numerical analysis software* (Wikipedia, 2012b). El primero ofrece un conjunto de “*tablas que proporcionan una comparación de los software para análisis numérico*”. Se hace una primera división en categorías: aplicaciones y librerías². Para las veintiocho (28) aplicaciones analizan: características generales (creador, costos, licencia, última versión, entre otros) y sobre cuáles Sistemas Operativos puede ser ejecutado (Windows, Mc OS X, Linux, entre otros). El segundo, es una lista de “*aplicaciones informáticas destinadas a su uso con el análisis numérico o análisis de datos*”³. En este caso, el autor las clasifica en MATLAB y sus clones, Sistemas de Álgebra Computacional (o CAS), Interfaces orientadas y Lenguajes orientados.

A partir de la información extraída de los sitios mencionados en el párrafo anterior, se comenzó un análisis de fortalezas y debilidades que tiene nuestra aplicación frente a las ofertas existentes. Se tuvieron en cuenta los siguientes criterios para identificar los puntos fuertes y débiles:

- **Disponibilidad/Accesibilidad.** Aquí se analizaron las posibilidades que tienen estudiantes y docentes de acceder y utilizar la aplicación.
- **Usuarios.** En este punto se tuvo en cuenta hacia quién estaba diseñado el software, lo que impacta directamente sobre los conocimientos previos que se debe tener para utilizar la aplicación.
- **Tipo de software.** En este punto se intentó corroborar si la aplicación se diseñó con alguna intencionalidad pedagógica o simplemente para el desarrollo de cálculos matemáticos.

CONSIDERACIONES FINALES

En el sitio que ofrece las comparativas, de las veintiocho (28) aplicaciones sólo doce (12) son gratuitas (Licencia GPL⁴). De las que son comerciales, el costo va desde U\$D 100 hasta U\$D 2800; de éstos, cuatro lo ofrecen libre para estudiantes. Del total de aplicaciones comparadas, sólo siete (7) se centran en el cálculo numérico (NCS) y de ellas, cinco son libres y gratuitas.

El otro sitio, con la lista de aplicaciones, de sesenta y siete (67) aplicaciones que se listan, nueve (9) son específicas para el cálculo numérico (NCS), hay cuatro (4) interfaces orientadas y cuatro (4) lenguajes que abordan algún aspecto del cálculo numérico. El resto, están centrados en el análisis de datos o son CAS. Esto da un total de dieciseis (16) recursos asociados al cálculo numérico de los cuales once (11) son libres y seis (6) pagos.

Como se puede apreciar, de los párrafos anteriores y a la luz de los criterios usados en este trabajo, muchas de las aplicaciones limitan su utilización por ser propietarias y si bien son, en general, compatibles con el SO Windows™, exigen de los usuarios un proceso de instalación. Nuestra aplicación está disponible en la Web de manera que es accesible de forma libre y gratuita por cualquier *internauta* y compatible con todos los sistemas operativos; sólo hace falta un navegador. Esto también implica que no es posible utilizarla fuera de línea, lo cual exige que el estudiante/docente cuente con una conexión a Internet.

Por otra parte, cada aplicación cuenta con la impronta de su diseñador, por lo tanto el estudiante/docente debe invertir un tiempo de aprendizaje de la aplicación –más allá de sus conocimientos sobre Cálculo Numérico. Nuestra aplicación se utiliza a través de un

² Conjunto de subprogramas utilizados para desarrollar software.

³ El análisis de datos ofrece metodologías y estrategias que permiten inspeccionar, transformar y modelizar los datos con el objetivo de resaltar información útil, inferir conclusiones y apoyar la toma de decisiones.

⁴ La Licencia Pública General es una licencia creada por la *Free Software Foundation* y fue diseñada para proteger la libre distribución, modificación y uso de software (<http://www.gnu.org/copyleft/gpl.html>).

navegador; además, todas las opciones que se proponen son a partir de enlaces y los datos se completan sobre cuadros de texto o listas desplegables. Esto hace que los estudiantes/docentes cuenten con los conocimientos básicos –los de cualquier *internauta*– para operar la aplicación, con lo cual puede centrar su atención en los contenidos.

Como se puede apreciar en los primeros párrafos, tanto la lista de aplicaciones como la comparación ofrecen un conjunto de NCS, pero en ningún caso se habla de MES. Creemos que esta es una de las principales ventajas que ofrece nuestra aplicación, ya que fue diseñada con una intencionalidad pedagógica y no sólo para la resolución de problemas matemáticos.

REFERENCIAS

- ASCHERI, M. E., PIZARRO, R., ASTUDILLO, G., GARCÍA, P. y CULLA, M. E. 2007. *Relevamiento de software en línea para la enseñanza-aprendizaje de métodos numéricos. Herramientas para su desarrollo*. V CIEMAC. Cartago. Costa Rica, 20-24. Disponible en <http://www.cidse.itcr.ac.cr/ciemac/>
- ASCHERI, M. E., PIZARRO, R., GARCÍA, P., ASTUDILLO, G. y CULLA, M. E. 2008a. *Un software educativo con herramientas libres y acceso Web para temas de Cálculo Numérico: un primer prototipo*. II REPEM. Santa Rosa. La Pampa. Argentina, 223-230.
- ASCHERI, M. E., PIZARRO, R., ASTUDILLO, G., GARCÍA, P. y CULLA, M. E. 2008b. *Métodos de resolución de ecuaciones no lineales con software elaborado con herramientas libres y acceso Web*. 5ª Jornada de Informática y Educación. Villa María. Córdoba. Argentina, 1-11. Disponible en <http://jornadaie.unvm.edu.ar/>
- ASCHERI, M. E., ASTUDILLO, G., GARCÍA, P., PIZARRO, R. y CULLA, M. E. 2009. *Análisis de un software educativo para Cálculo Numérico*. VI CIEMAC. Cartago. Costa Rica, 148-158. Disponible en <http://www.cidse.itcr.ac.cr/ciemac/>
- ASCHERI, M. E., ASTUDILLO, G., GARCÍA, P., PIZARRO, R. y CULLA, E. 2010. *Elaboración de un software educativo usando herramientas gratuitas. Primeras EVALUACIONES*. WICC 2010. EL CALAFATE. SANTA CRUZ. ARGENTINA, 734-738.
- ASCHERI, M. E., PIZARRO, R. A., ASTUDILLO, G. J., GARCÍA, P. y CULLA, M. E. 2011. *Software educativo desarrollado para temas de Cálculo Numérico: Últimos avances*. XVI EMCI Nacional y VIII EMCI Internacional. Olavarría. Argentina, 1-8.
- BALDERAS PUGA, A. 1999. *The influence of information technology in the daily work of mathematics teachers*. International Conference on Mathematics Education into the 21st Century: Societal Challenges, Issues and Approaches, 3, 1-8.
- BOSCO, A. 2002. *Los recursos informáticos en la escuela de la sociedad de la información: deseo y realidad*. Revista EDUCAR, 29, 125-144.
- CAMILLONI, A. 1995. *El tratamiento del error en situaciones de baja interacción y respuesta demorada*. Revista Rueda, 2(2), 30-36.
- CATALDI, Z. 2000. *Metodología de diseño, desarrollo y evaluación de software educativo* (Tesis de Magister en Informática, UNLP).
- Comparison of numerical analysis software. 2012a, Marzo 27. In *Wikipedia, The Free Encyclopedia*. Recuperado 20:52, Mayo 9, 2012, desde http://en.wikipedia.org/w/index.php?title=Comparison_of_numerical_analysis_software&oldid=484184443
- GÓMEZ SÁNCHEZ, R. 2004. *Software libre vs. Software propietario: Programando nuestro futuro*. HAOL, 2, 125-140. Recuperado 20:30, Mayo 9, 2012, desde <http://www.cidse.itcr.ac.cr/ciemac/5toCIEMAC/Ponencias/Relevamientodesoftwareonlinea.pdf>

- List of numerical analysis software. 2012b, Marzo 22. In *Wikipedia, The Free Encyclopedia*. Recuperado 20:49, Mayo 9, 2012, desde http://en.wikipedia.org/w/index.php?title=List_of_numerical_analysis_software&oldid=483363620
- LOSADA LISTE, R. 2007. *GeoGebra: la eficiencia de la intuición*, *Gaceta de la Real Sociedad Matemática Española*, 10 (1), 223-240.
- MARQUÉS, P. 1996. *El software educativo*. Universidad Autónoma de Barcelona. Consultado en Febrero, 8, 2010 en http://www.lmi.ub.es/te/any96/marques_software
- QUINTANA MENDOZA, J. y MENDOZA BOLO, M. 2006. *Guía para el desarrollo del pensamiento a través de la matemática*. Perú: Fimart S.A.C.
- RODINO, A. M. 1996. *Las nuevas tecnologías informáticas en la educación: viejos y nuevos desafíos para la reflexión pedagógica*. En *Memorias del VII Congreso Internacional sobre Tecnología y Educación a Distancia*, 51-71. Costa Rica: EUNED.
- SANTANDREU PASCUAL, M. 2004. *Recursos TIC en la enseñanza y aprendizaje del área de matemáticas*. *Comunicación y pedagogía: Nuevas tecnologías y recursos didácticos*, 200, 65-70.
- SQUIRES, D. y MC DOUGALL, A. 1994. *Choosing and using educational software: a teachers' guide*. London: The Falmer Press.
- VILLEGAS, F. 2007. TIC y Matemáticas. *Unión: revista iberoamericana de educación matemática*, 9, 149-163. Disponible en <http://www.fisem.org/paginas/union/info.php?id=186>
- ZHAO, Y. 1998. *Blind Trust in Authentic Mathematical Tools in Mathematics Education*. *The International Journal of Computers Algebra in Mathematics Education*, 3, 161-173.