

# Aspectos computacionales en la estimación de incertidumbres de ensayo por el Método de Monte Carlo

## Computational aspects in uncertainty estimation by Monte Carlo Method

Constantino, Pablo <sup>(1)</sup>

<sup>(1)</sup> Departamento de Metrología Física, Laboratorio Tecnológico del Uruguay, LATU.

Contacto: pconstan@latu.org.uy

Recibido: 18/4/2013 - Aprobado: 5/12/2013

### Resumen

El propósito de este trabajo es analizar los distintos aspectos relacionados al desarrollo de una aplicación informática para la estimación de incertidumbres de ensayo por el método de Monte Carlo, independiente de plataformas de cálculo como MS Excel, MathLab o R. Se analizan las dificultades y posibles soluciones en cada una de las etapas necesarias para alcanzar este objetivo, el algoritmo para la creación de un intérprete de ecuaciones, la generación de números pseudo-aleatorios con las distribuciones de probabilidad más frecuentes y el tratamiento de incertidumbres Tipo A por este método.

Finalmente se hace un estudio comparativo de los resultados obtenidos con la aplicación generada, el método clásico (GUM) y la misma simulación realizada con el Software R. Este estudio se realiza sobre el cálculo de la densidad del aire según ecuación CIPM, la presión generada por una balanza de presión y la estandarización de una solución de hidróxido de sodio de acuerdo al ejemplo A2 de la guía EURACHEM / CITEC CG 4.

Palabras clave: Metrología, incertidumbre, Monte Carlo, GUM, software, código fuente, Delphi.

### Abstract

The purpose of this paper is to analyze the various aspects related to the development of a software application aimed to uncertainty estimation by Monte Carlo Method, independent from worksheet or third party applications, such as MS Excel, MathLab or R. Difficulties and their available solutions around the needed stages to achieve this goal are discussed, as well as the algorithms needed for creating a math equations parser, the pseudo-random numbers generators with main probability distributions found in uncertainty calculations, and the management of Type A uncertainties.

The application developed is tested on three samples: Air density determination according to CIPM equation, the pressure generated by a Pressure Balance and the standardizing of a solution of sodium hydroxide as in example A2 of EURACHEM/CITEC CG 4 guide. Finally, these results are compared with the results obtained applying the classical method (GUM) and those obtained using R software.

Keywords: Metrology, Uncertainty, Monte Carlo, Software, GUM, source code, Delphi.

## Introducción

Si bien existen desde hace varios años numerosas aplicaciones informáticas para la realización de simulaciones de Monte Carlo en la resolución de diferentes problemas físicos, matemáticos o de negocios, es a partir de la aprobación del Suplemento 1 a la Guía de la Expresión de la Incertidumbre de Medida (JCGM 101, 2008) cuando comienza a popularizarse este método como una alternativa al sistema clásico (GUM) de estimación de incertidumbres de ensayo o para la validación de los resultados obtenidos por este método.

### Tecnología

La tecnología más popular en las aplicaciones de software disponible para la realización de simulaciones de Monte Carlo

en modelos matemáticos está sustentada en el motor de cálculo de MS Excel, distribuyéndose como complemento de este paquete de software. Este enfoque particular de las aplicaciones implica la adaptación de los modelos matemáticos de los ensayos al formato de la hoja de cálculo, lo que impide el tratamiento de una o más ecuaciones independientes, en la forma que trabajan otras aplicaciones informáticas reconocidas para el cálculo de incertidumbres según la perspectiva GUM, por ejemplo, GUM Workbench®.

Por otra parte, este formato requiere la realización de operaciones previas a la simulación, como el cálculo del desvío estándar para contribuyentes de incertidumbre tipo A a partir de un número determinado de datos experimentales, debido a que en general no es posible el tratamiento de datos experimentales en forma independiente sino a partir de distribuciones de probabilidad específicas.

En este trabajo se analizan los aspectos requeridos para desarrollar una aplicación independiente de plataformas externas de cálculo utilizando un lenguaje de programación de propósito general, que permita realizar simulaciones de Monte Carlo sobre modelos matemáticos de ensayos. Por otra parte se diseñan funciones que

permiten el manejo de contribuyentes de incertidumbre tipo A y tipo B de la misma forma que se trabaja habitualmente en el método GUM (JCGM 100, 2008).

basado en el MMC, para cuantificar el impacto de la incertidumbre de cada contribuyente en la incertidumbre final de la magnitud de salida.

### MMC vs GUM

Existen impedimentos y dificultades conocidas en el tratamiento de incertidumbres en modelos matemáticos complejos según la plataforma GUM. Entre los impedimentos se encuentra, por ejemplo, el caso en que las funciones de probabilidad (FDP) de las magnitudes de entrada sean asimétricas. Pero también se hace inconveniente el enfoque GUM cuando existe dificultad en la obtención de derivadas parciales para el cálculo de coeficientes de sensibilidad para los diferentes contribuyentes, o el cálculo del número efectivo de grados de libertad a partir de la ecuación de Welch y Satterthwaite cuando algunas variables del modelo tienen grados de libertad de valor finito.

A pesar de esto, el método GUM permite obtener información valiosa para el mejoramiento de la performance de los ensayos y que en muchos casos no puede ser obtenida en forma directa por medio del método de Monte Carlo.

Un ejemplo es la evaluación de incertidumbres estándar de las magnitudes de entrada, que da la posibilidad al analista de enfocar la mejora de los procesos en variables que impacten significativamente en la incertidumbre total del mesurando. En el Método de Monte Carlo (MMC) se obtiene como datos de salida una colección de resultados cuya distribución es una representación de la FDP del mesurando, quedando los aportes de incertidumbre de cada magnitud del modelo dentro de una “caja negra” de datos de entrada.

Se estudia también en este trabajo un procedimiento adicional

### Funciones de cálculo

Dentro de los diferentes aspectos que deben considerarse, no solamente se encuentran los de tipo estadístico y/o metrológico como los mencionados, sino también un gran número de tareas de tipo general, como la realización de un intérprete de ecuaciones, la obtención de un intervalo de confianza para el mesurando o la necesaria optimización del tiempo de ordenamiento de resultados de la simulación, ya que el tiempo implicado en este proceso, realizado sobre cientos de miles de resultados posibles del mesurando, podría insumir inconvenientemente varios minutos de proceso en el computador si se utilizan linealmente los algoritmos básicos de ordenamiento de datos.

Finalmente, en el desarrollo de la aplicación también se abordan tareas referidas específicamente al método de simulaciones, por ejemplo la selección de algoritmos adecuados en la generación de números pseudo-aleatorios.

Para los fragmentos de código incluidos en este trabajo se utiliza el Lenguaje Object Pascal. Las pruebas de simulación y aplicaciones resultantes se diseñan en plataforma de Delphi 7.

### Materiales y Métodos

Se toma como esquema básico de la aplicación el diagrama de la Figura 1 y se analizan las diferentes metodologías y algoritmos que pueden utilizarse en cada etapa del proceso.

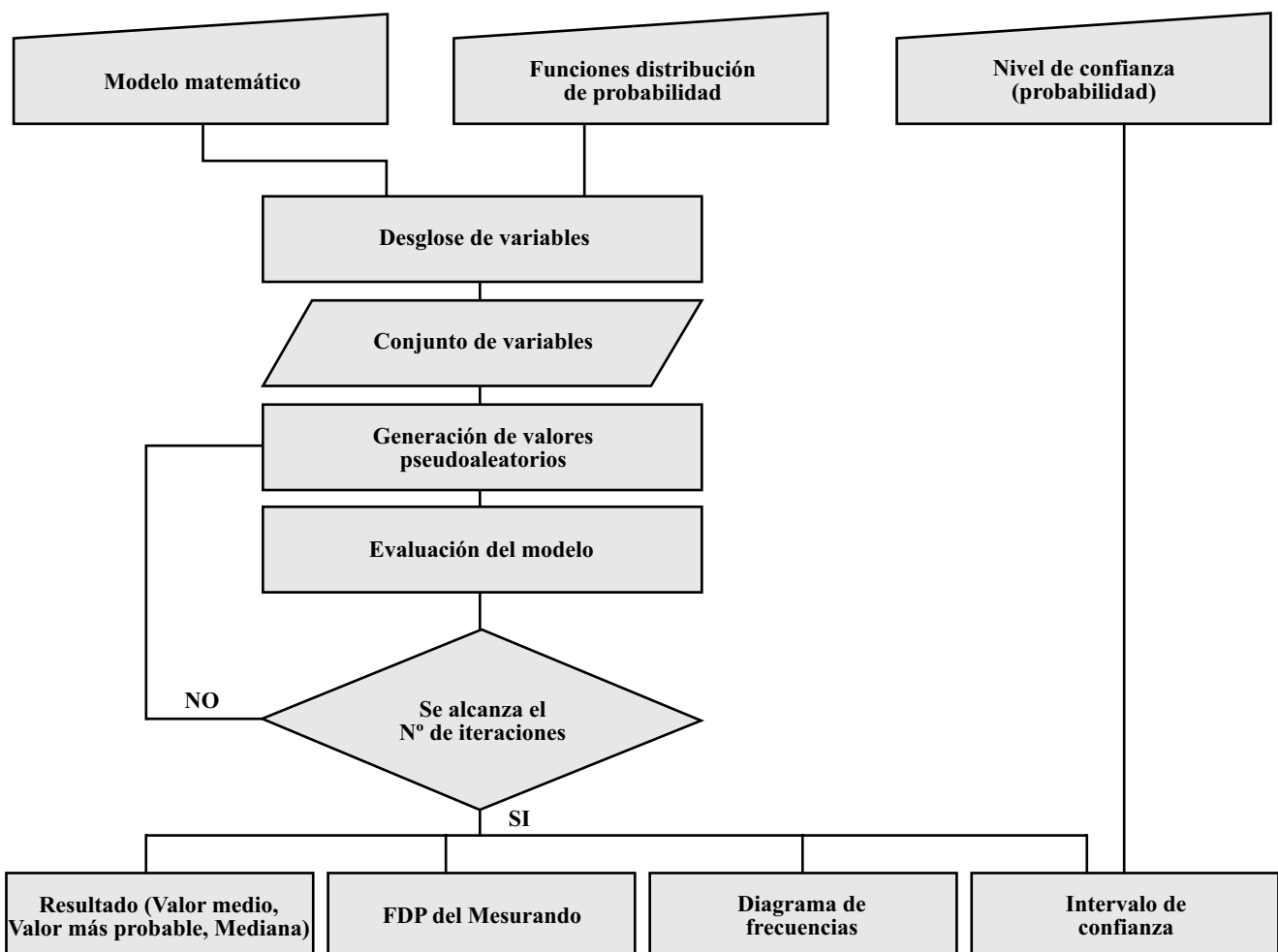


Figura 1. Representación básica del flujo de la aplicación.

## Algoritmos de tipo general

**Intérprete de ecuaciones.** Para la realización de esta tarea se convierte previamente la notación común del modelo en notación postfija, llamada también polaca inversa o RPN (Burks et al., 1954) diseñada específicamente para la optimización de recursos informáticos en la evaluación de expresiones.

Este esquema de notación algebraica permite que los cálculos del modelo se realicen en forma secuencial y sin reglas de preferencia para los operadores, mediante la utilización de una estructura informática de pila (Stack), esto es, una lista de datos cuyo acceso implica que los últimos ítems en entrar a la pila serán los primeros en salir. La notación corriente para expresar ecuaciones (llamada infija) es entonces convertida previamente a postfija o RPN para su tratamiento.

**Traductor de notación infija a RPN o postfija.** Partiendo, entonces, del modelo matemático presentado como una sucesión de ítems (variables, constantes, operadores y paréntesis), las etapas para traducir entre notaciones infija y postfija están detalladas en el algoritmo Shunting Yard esquematizado en la Figura 2.

centrada. La mayoría de los lenguajes de programación actuales poseen funciones prediseñadas para obtener números pseudoaleatorios que se ajusten a estas distribuciones, o al menos para distribuciones normales y uniformes. Sin embargo, aun para estos casos, se proponen regularmente diferentes enfoques al problema de la generación de aleatorios y nuevos algoritmos que permiten mejorar la calidad de la secuencia de los números obtenidos.

Para evaluar el impacto del algoritmo utilizado en la generación de números para la simulación en los resultados de incertidumbre, se realiza un análisis comparativo realizando 50 simulaciones con diferentes algoritmos y se evalúan las diferencias de media e intervalo de confianza para 95% de probabilidad en 200 mil repeticiones.

El caso de la distribución uniforme se da cuando de la magnitud  $X$  se conocen solamente un límite inferior ( $a$ ) y uno superior ( $b$ ). Esto es, por ejemplo, el caso de la indicación de un equipo digital, donde la resolución del equipo determina el intervalo de valores entre los cuales se encuentra la magnitud (con igual probabilidad) en el dominio de los reales. Para distribución uniforme se utiliza la función disponible en Delphi 7, y el algoritmo Mersene Twister (Matsumoto y Nishimura, 1998) en versión Object Pascal (Krejci, 1999).

Se van tomando secuencialmente los ítems de la expresión infija	
ÍTEM	ACCIÓN
"("	Se agrega a la Pila
")"	Se extrae secuencialmente ítems de la pila y agregarlos a ecuación postfija, hasta encontrar "(" que se deja en la pila
Operador	Si la pila está vacía se agrega a la Pila Si la jerarquía del ítem es mayor que la del tope de la pila, se agrega a la Pila De lo contrario, se extrae el ítem al tope de la pila y se agrega a la ecuación postfija repitiendo la comparación con el nuevo tope
Operando	Se agrega a ecuación postfija
Al final de la expresión extraer y desplegar los elementos de la pila en la ecuación postfija hasta que se vacíe.	

Figura 2. Algoritmo Shunting Yard.

Para la evaluación de la ecuación resultante en notación RPN se utiliza el algoritmo ilustrado en la Figura 3.

ÍTEM	ACCIÓN
Operando	Se agrega a la Pila
Operador	Tomar los últimos dos valores de la pila y aplicarles el operador. El resultado de la operación se sustituye por éstos en la pila

Figura 3. Representación del flujo de cálculo de ecuaciones en notación RPN.

**FDP y algoritmos para generación de números pseudoaleatorios.** Según las características del mesurando, pueden existir en el modelo de entrada magnitudes con diversas distribuciones de probabilidad. En este trabajo se estudian algoritmos para las distribuciones de probabilidad más comunes en los trabajos de ensayo y calibración: normal (gaussiana), rectangular (uniforme) y triangular

Cuando la información disponible sobre una magnitud  $X$  es su mejor estimación y su incertidumbre típica asociada  $u(x)$ , puede asumirse que  $X$  posee una distribución de probabilidad gaussiana o normal. Esta distribución puede utilizarse cuando los datos provienen de un certificado de calibración con factor de cobertura  $k=2$ . Si bien la guía JCGM 101:2008 sugiere para estos casos la distribución  $t$ , para grados de libertad  $GL \geq 30$  (caso en que  $k=2$ ), la distribución  $t$  puede aproximarse a una distribución gaussiana. En distribución normal se comparan resultados obtenidos con el algoritmo de Marsaglia y Bray (1964), y el algoritmo básico de Box y Müller (1958) (Thomas et al., 2007).

Cuando una magnitud  $X$  está definida como la suma de dos magnitudes independientes con distribución de probabilidad uniforme, con el mismo intervalo de amplitud, se obtiene una distribución triangular centrada. En algunos casos, esta distribución puede asignarse convenientemente a lecturas obtenidas en un instrumento analógico, donde la semiamplitud de la distribución triangular será igual a la estimación visual dentro de la división del instrumento. Para distribuciones triangulares se comparan resultados por el método de Transformación Inversa y el de convolución (Saucier, 2000).

```

Function Rand_TriangularTInv(min,max,mode: Extended):
Double;
var
  u, r: Double;
begin
  u := Random;
  if u <= (mode-min)/(max-min) then r := min+sqrt(u*(max-
min)*(mode-min))
  else r := max-sqrt((1-u)*(max-min)*(max-mode));
  Result := r;
end;

Function Rand_TriangularConv(min,max: Extended): Double;
var
  u, p, r: Double;
begin
  u := Random*(max-min)/2;
  p := Random*(max-min)/2;
  r := u+p+min;
  Result := r;
end;
    
```

Figura 4. Ejemplo de funciones generadoras de variables aleatorias con distribución triangular por transformada inversa y convolución.

**Intervalo de confianza.** Con el fin de obtener el intervalo de confianza en el cálculo de incertidumbre, es necesario ordenar los datos de forma tal que sea posible determinar los valores que acotan un porcentaje de los resultados obtenidos, de valor igual a la probabilidad de cobertura.

Lo que se busca en este caso es únicamente obtener las cotas del intervalo de confianza, por tanto se minimiza el tiempo de proceso, ordenando solamente los datos correspondientes a las probabilidades de rechazo.

Como ejemplo, en el caso de un 95% de probabilidad, el intervalo de confianza queda determinado por el último valor del 2,5% menor de la población de resultados, y el primer valor del 2,5% mayor. Por esto, solo es necesario ordenar una fracción mínima de los resultados obtenidos en la simulación. Este método permite reducir sensiblemente el tiempo de procesamiento, por lo que pueden utilizarse algoritmos de sustitución simple u otro algoritmo básico sin afectar en forma significativa el rendimiento de la simulación.

**Incertidumbres tipo A.** La mayoría de las aplicaciones para Monte Carlo disponen un número determinado de distribuciones de probabilidad que pueden ser asignadas a los diferentes contribuyentes del modelo. En el caso de contribuciones de tipo estadístico esta forma de trabajo deja al analista un necesario tratamiento previo de los datos.

Para esta aplicación se diseña una interfase con el usuario que permita la entrada de datos experimentales de forma manual y luego los convierte en forma autónoma a valores de distribución normal u otra deseada, mediante el cálculo de sus parámetros estadísticos (media y varianza).

Cuando los grados de libertad del contribuyente de incertidumbre tipo A son  $GL \geq 30$ , puede atribuirse una distribución normal. A partir de estos valores calculados es obtenida la media y el desvío estándar, atribuyéndose finalmente al contribuyente una distribución normal con los parámetros calculados, en el supuesto de que los grados de libertad tiendan a valores infinitos.

La Guía JCGM 100:2008 indica que la FDP asociada a una serie de indicaciones (contribuyente tipo A) es la distribución t de Student. Por lo que en caso que  $GL = n^\circ$  de valores -  $1 < 30$ , la simulación más adecuada se realiza según esta distribución. Para este caso, se utiliza el algoritmo basado en el método de Kinderman y Monahan propuesto en lenguaje Fortran por John Dagpunar (Dagpunar, 1988). La Figura 5 es una traducción a lenguaje Object Pascal del algoritmo mencionado.

```

function Rand_t (const GL: Integer; sesgo: Double): Extended;
{ La variable sesgo permite que la simulación no esté centrada en
el valor cero }
var
  s, c, a, f, g: Extended;
  r, x, v: Extended;
  Done: Boolean;
begin
{ Debe controlarse previamente que GL no sea < 1 }
  s := GL;
  c := -0.25 * (s + 1.0);
  a := 4.0 / Power (1.0 + 1.0 / s, c);
  f := 16.0 / a;
  if (GL > 1) then begin
    g := s - 1.0;
    g := Power ((s + 1.0) / g, c) * Sqrt ((s + s) / g);
  end else g := 1.0;
  Done := False;
  x := 0;
  repeat
    r := Random;
    if (r <= 0.0) then Continue;
    v := Random;
    x := (2.0 * v - 1.0) * g / r;
    v := Sqr (x);
    if (v > 5.0 - a * r) then begin
      if (GL >= 1) and (r * (v + 3.0) > f) then Continue;
      if r > Power (1.0 + v / s, c) then Continue;
    end;
    Done := True;
  until Done;
  Result := sesgo + x;
end;
    
```

Figura 5. Código para generación de números aleatorios con distribución t de Student.

**Incertidumbres estándar.** Como se aludió anteriormente, el cálculo de incertidumbres estándar de cada contribuyente es un dato de salida inherente al método GUM, que no está disponible si la estimación se realiza por medio del método de Monte Carlo.

Sin embargo, el valor de incertidumbre estándar permite al analista conocer las magnitudes controlantes de su proceso de ensayo y por tanto enfocar en éstas las mejoras de la calidad del mismo, por lo que es una especificación valiosa para cualquier aplicación que permita el cálculo de incertidumbres.

En el caso del Monte Carlo, una medida del impacto que tiene la incertidumbre de estos contribuyentes sobre la incertidumbre total del modelo puede estimarse a través de una secuencia de simulaciones sobre cada una de las variables manteniendo, en cada caso, el resto de las componentes del modelo como constantes, de valor igual a su esperanza matemática. Para obtener estos valores con una precisión razonable, se realizan un mínimo de 1000 iteraciones sobre cada magnitud de entrada.

## Resultados y Discusión

Para la evaluación y comparación de los distintos aspectos analizados en el presente documento, se utiliza el concepto de tolerancia asociada a un valor numérico,  $\delta$ , según recomienda la Guía JCGM 101:2008. Para la obtención de este parámetro, se parte del valor numérico  $z$ , del que se quiere conocer su tolerancia asociada. Se expresa  $z$  en forma:  $c \times 10^l$ , donde  $c$  es un número entero con un número determinado de cifras significativas ( $n$ ) y  $l$  un número entero.

Por ejemplo: un valor de 0,00072 se expresa como  $72 \times 10^{-5}$ , donde  $n$  será 2 (ambas son cifras significativas) y  $l$  será -5. La tolerancia numérica se obtiene entonces a partir de la ecuación:

$$\delta = \frac{1}{2} 10^l$$

En el caso particular de la expresión de incertidumbres, la tolerancia numérica más exigente, esto es, el caso más desfavorable, se dará para valores de  $99 \times 10^l$ , la que será de  $0,5 \times 10^l$ . De este modo, un valor  $m$  tal que:

$$m \leq \frac{0,5 u(y)}{100}$$

se considera dentro de la tolerancia numérica para cualquier valor de incertidumbre.

Para las comparaciones siguientes, se obtienen valores porcentuales de impacto frente a valores de incertidumbre, tomando como límite de aceptación 0,5%, según ecuación anterior.

**Número de iteraciones.** Para estimar el impacto del número de iteraciones en el cálculo de incertidumbre por simulación se realizan 100 simulaciones en 20000, 100000, 200000 y 1 millón de repeticiones, para una variable aleatoria con distribución uniforme de valor medio=1 y semiintervalo=1.

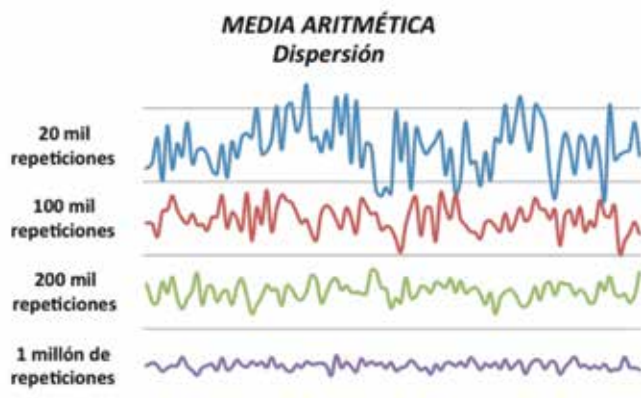


Gráfico 1. Dispersión comparativa de la media aritmética obtenida en 100 simulaciones para 20 mil, 100 mil, 200 mil y 1 millón de repeticiones.

El impacto de esta dispersión de medias e incertidumbres para una probabilidad de cobertura de 95%, relativas al valor de incertidumbre informado, se muestra en el Gráfico 2. Los valores porcentuales del gráfico corresponden al semiintervalo de amplitud de esta dispersión, relativas a la incertidumbre estimada para un 95% de probabilidad.

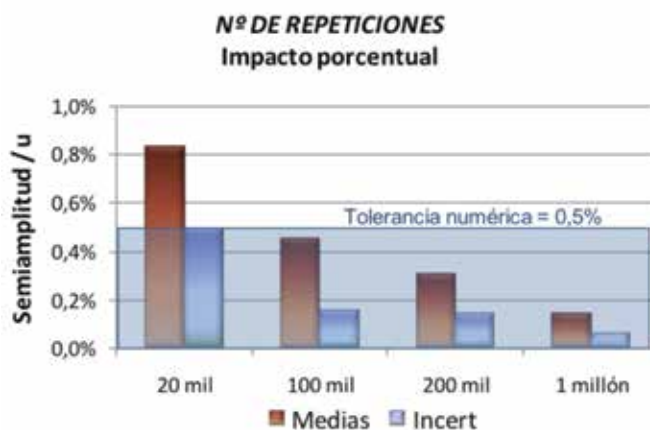


Gráfico 2. Dispersión porcentual de las medias encontradas y los semiintervalos de confianza, calculadas como  $((\text{máx}-\text{mín})/2)/u(y)$  para 95% de probabilidad de cobertura.

El gráfico muestra que para 20 mil iteraciones, la dispersión obtenida en las medias aritméticas obtenidas puede ser sensiblemente mayor que la tolerancia numérica de algunos valores de incertidumbre. A partir de 100 000 esta dispersión no es significativa, ya que presenta valores menores a la tolerancia numérica de cualquier valor de incertidumbre.

**Comparación de algoritmos generadores de números aleatorios.** Se realizan 50 simulaciones según diferentes algoritmos para 200 000 iteraciones, comparando los valores de media e incertidumbre encontrados para un 95% de confianza. Los gráficos siguientes muestran el impacto porcentual de estas diferencias, calculados como  $(\text{algoritmo}_A - \text{Algoritmo}_B)/u(y)$  para 95% de probabilidad de cobertura.

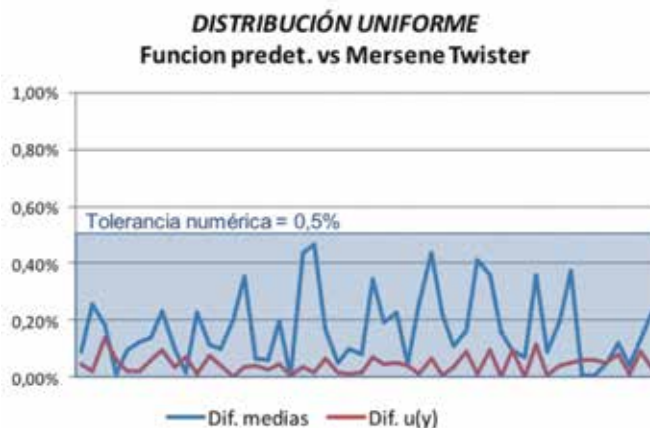


Gráfico 3. Impacto porcentual en el uso de diferentes algoritmos generadores de aleatorios uniformes.

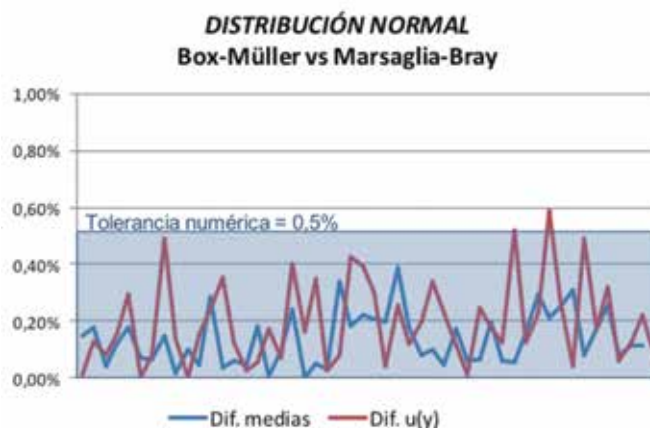


Gráfico 4. Impacto porcentual en el uso de diferentes algoritmos generadores de aleatorios con FDP Normal.

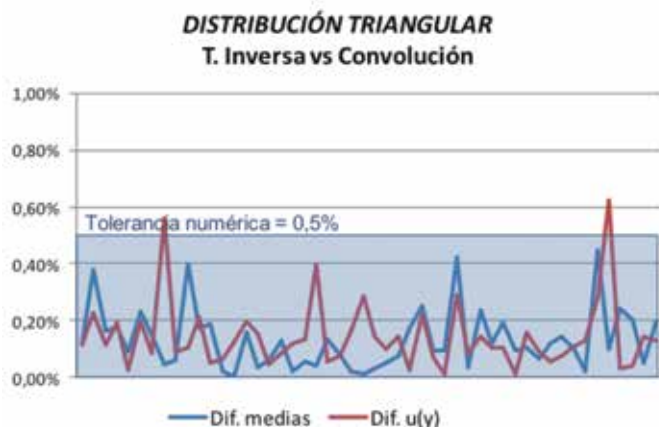


Gráfico 5. Impacto porcentual en el uso de diferentes algoritmos generadores de aleatorios con FDP Triangular.

En los Gráficos 3, 4 y 5 puede verse que no existen diferencias significativas en los resultados debidas al uso de diferentes algoritmos para la generación de variables aleatorias con distribución uniforme, normal o triangular, en simulaciones aplicadas a la estimación de incertidumbres. Puede notarse sin embargo, que existe una probabilidad no nula de que la diferencia entre medias o incertidumbres para ambos algoritmos estudiados en algunas distribuciones tome un valor similar o incluso mayor que la tolerancia numérica mínima. No obstante, aun en las máximas diferencias encontradas, la interpretación de estos valores es que arrojarían resultados incompatibles solamente en los casos más desfavorables y nunca mayor que una unidad en la segunda cifra significativa de la incertidumbre.

Por tanto, esta razonable homogeneidad en los resultados para el caso de la estimación de incertidumbres por el método de Monte Carlo permite inferir que la búsqueda de mejores algoritmos no aportará mejoras significativas al método, por lo que el uso de algoritmos predeterminados es recomendado.

**Cálculos con la aplicación generada.** Se comparan los resultados obtenidos con la aplicación generada y el Método GUM.

Con:

**$t$  : temperatura ambiente**

**$h$  : humedad relativa**

**$\alpha$  : 1,0062**

**$\beta$  :  $3,14 \times 10^{-8} \text{ Pa}^{-1}$**

**$\gamma$  :  $5,6 \times 10^{-7} \text{ k}^{-2}$**

**$A$  :  $1,2378847 \times 10^{-5} \text{ k}^{-2}$**

**$B$  :  $-1,9121316 \times 10^{-2} \text{ k}^{-1}$**

**$C$  : 33,93711047**

**$D$  :  $-6,3431645 \times 10^{-3} \text{ K}$**

## Ejemplo 1.

### Densidad del aire según CIPM-2007

Se realiza el cálculo de densidad del aire según la fórmula propuesta en CIPM-2007. La ecuación para el cálculo de densidad del aire viene dada por la ec:

$$\rho = \frac{p \cdot M_a}{Z \cdot R \cdot T} \left[ 1 - x_v \cdot \left( 1 - \frac{M_v}{M_a} \right) \right]$$

Con:

$\rho$  : densidad del aire en kg/m<sup>3</sup>

$p$  : presión atmosférica en Pa

$M_a$  : masa molar de aire seco =  $28,96546 \cdot 10^{-3} \text{ kg/mol}$

$Z$  : factor de compresibilidad adimensional

$R$  : constante molar de los gases.

$T$  : temperatura en K.

$x_v$  : fracción molar de vapor de agua adimensional

$M_v$  : masa molar de vapor de agua  $0,01801528 \text{ kg/Mol}$ .

El valor de  $x_v$  se determina a partir de:

$$x_v = \frac{h}{100} (\alpha + \beta \cdot p + \gamma \cdot t^2) \cdot \frac{e^{(AT^2+BT+C+\frac{D}{T})}}{P}$$

Mientras que  $Z$  está determinada por la fórmula:

$$Z = 1 + \frac{P}{T} [a_0 + a_1 t + a_2 t^2 + (b_0 + b_1 t)x_v + (c_0 + c_1 t)x_v^2] + \frac{p^2}{T^2} (d + ex_v^2)$$

**$a_0 = 1,58123 \times 10^{-6} \text{ KPa}^{-1}$**

**$a_1 = -2,9331 \times 10^{-8} \text{ Pa}^{-1}$**

**$a_2 = 1,1043 \times 10^{-10} \text{ K}^{-1} \text{ Pa}^{-1}$**

**$b_0 = 5,707 \times 10^{-6} \text{ KPa}^{-1}$**

**$b_1 = -2,051 \times 10^{-8} \text{ Pa}^{-1}$**

**$c_0 = 1,9898 \times 10^{-4} \text{ KPa}^{-1}$**

**$c_1 = -2,376 \times 10^{-6}$**

**$d = 1,83 \times 10^{-11} \text{ k}^2 \text{ Pa}^{-2}$**

**$e = -0,765 \times 10^{-8} \text{ K}^2 \text{ Pa}^{-2}$**

La representación en cadena de texto para utilizar en la aplicación generada en este trabajo puede escribirse según el siguiente grupo de ecuaciones:

```

Rho_a=((P*Ma)/(Z*R*(273.15+Temp))*(1-xv*(1-(Mv/Ma)))
xv=H*0.01*f*Psv/P
Psv=EXP(A*(273.15+Temp)^2+B*(273.15+Temp)+C+D/(273.15+Temp))
f=alfa+beta*P+gama*Temp^2
Z=1-(P/(273.15+Temp))*(a_0+a_1*Temp+a_2*Temp^2+(b_0+b_1*Temp)*xv+
(c_0+c_1*Temp)*xv^2)+(P^2/(273.15+Temp)^2)*(d+e*xv^2)
Temp=Temp_cal+Temp_div
P=P_cal+P_div
H=H_cal+H_div
    
```

En estas ecuaciones las magnitudes de temperatura, humedad y presión aportan la incertidumbre debida a su calibración ( $\_cal$ ) e incertidumbres por resolución de los instrumentos de medida ( $\_div$ ). Estos parámetros toman valor = 0, ya que es utilizada únicamente con fines de estimación de incertidumbre.

Datos de entrada:

Magnitud	Valor	Distrib	Param.
$Ma$	0.028965460	Uniforme	Semi intervalo = $1 \times 10^{-9}$
$R$	8.3144720	Uniforme	Semi intervalo = $1 \times 10^{-7}$
$Mv$	0.018015280	Uniforme	Semi intervalo = $1 \times 10E^{-9}$
$Temp\_cal$	21.2	Normal	Desvío standar = 0,065
$Temp\_div$	0	Uniforme	Semi intervalo = 0,05
$P\_cal$	101325.0164	Normal	Desvío standar = 149,5
$P\_div$	0	Uniforme	Semi intervalo = 5
$H\_cal$	50	Normal	Desvío standar = 1,7
$H\_div$	0	Uniforme	Semi intervalo = 0,05

Los demás parámetros de la ecuación son constantes, con el valor mencionado.

Resultados según método GUM:  $\rho = 1,1940 \pm 0,0036$  kg/m<sup>3</sup>

Resultado según MMC obtenido con la aplicación generada a partir de este trabajo:

Mediana: 1,19401 kg/m<sup>3</sup>; valor más probable: 1,1943 kg/m<sup>3</sup>; intervalo de confianza para 95,45% de probabilidad = [1,19045 ; 1,19758]

Según estos resultados:  $U_{MMC}(\rho) = (1,19758 - 1,19045)/2 = 0,003565$

Por lo que tomando como resultado para el MMC la mediana del conjunto de resultados se obtienen resultados idénticos para ambos métodos, observando una diferencia numérica para la incertidumbre de  $0,0036 - 0,003565 = 3,5 \times 10^{-5}$ .

La tolerancia numérica para el valor de incertidumbre según GUM (=0,0036) está dada por:

$$\delta = \frac{1}{2} 10^{-4} = 0,5 \times 10^{-4}$$

Por tanto,  $3,5 \times 10^{-5} < 0,5 \times 10^{-4}$  y los resultados por ambos métodos pueden considerarse compatibles, es decir, se obtiene un resultado idéntico según ambos enfoques, GUM y MMC.

## Ejemplo 2.

### Presión generada por una balanza de presión

Para este ejemplo se utiliza una balanza de presión de plataforma neumática, perteneciente al laboratorio de presión del LATU marca DHI, Modelo PG7102. El cálculo de incertidumbre se hará sobre la presión generada por un ensamble pistón-cilindro Marca DHI modelo PC-7100/7600-200 de constante nominal = 200 kPa/kg, para una presión nominal de 34,47 bar.

La presión generada en una balanza neumática es obtenida a partir del siguiente modelo:

$$P = \frac{M \cdot g_l \left(1 - \frac{\rho_a}{\rho_m}\right) + \sigma \cdot C}{A_0 \cdot (1 + \lambda P) \cdot [1 + \alpha \cdot (t - t_r)]} + (\rho_f - \rho_a) \cdot g_l \cdot \Delta h$$

Donde:

$M$ : masa total (nominal: 17,231 kg)

$g_l$ : gravedad local (en el laboratorio de presión del LATU  $g_l = 9,79732 \pm 1E-5$  m/s<sup>2</sup>)

$\rho_a$ : densidad del aire (se calculó a partir de la fórmula CIPM 2007 ( $\rho_a = 1,20048 \pm 0,0035$  kg/m<sup>3</sup>))

$\rho_m$ : densidad de las masas (según certificado)

$C$ : circunferencia nominal del pistón

$\sigma$ : coeficiente de tensión superficial del ensamble pistón cilindro ( $\sigma = \sigma_{pistón} + \sigma_{cilindro}$ )

$A_0$ : área efectiva a presión nula y temp. de referencia

$\lambda$ : coeficiente de deformación del ensamble pistón-cilindro con la presión.

$\alpha$ : coeficiente de deformación con la temperatura [el ensamble pistón/cilindro es de carburo de Tungsteno ( $\alpha = 4,5 \times 10^{-6} \text{ } ^\circ\text{C}^{-1}$ )].

El set de ecuaciones correspondiente a este modelo utilizado para el MMC es:

$$P_{ref} = \frac{((mp * (1 - \rho_a / \rho_p) + (mas * (1 - \rho_a / \rho_{as})) + SUM\_M * (1 - \rho_a / \rho_m)) * g + \sigma * C) / (A_o * (1 + (\alpha_c + \alpha_p) * (t - 20))) * (1 + \lambda * P)) / 100000 - ((\rho_f - \rho_a) * g * Ah) / 100000}{SUM\_M = G1 + G2 + G3 + D + B1 + M1 + N + P1 + Q + R}$$

Donde  $P_{ref}$  es la Presión generada por la balanza en bar.

Como puede verse, en el modelo para MMC se agregaron como datos de entrada independientes la masa del pistón ( $mp$ ) y su densidad ( $\rho_p$ ), así como los valores de masa y densidad para el asiento de masas (variables  $mas$  y  $\rho_{as}$ , respectivamente), dado que el certificado de las mismas indica masa real y densidad. El set de masas está calibrado en Masa convencional por lo que  $SUM\_M$  (en la segunda ecuación) es la suma de masas y la densidad ( $\rho_m$ ) será  $8000 \text{ kg/m}^3$  con incertidumbre = 0, por la propia definición de masa convencional.

En este modelo, las distribuciones de probabilidad para los valores de las masas, densidades, área específica ( $A_o$ ) y coeficiente de deformación ( $\lambda$ ) se tomaron como gaussianas con un valor medio igual al resultado indicado en el certificado y  $\sigma = u/k$ , debido a que según el certificado de calibración  $k=2$  en todos los casos.

Resultados según método GUM:  $P = 34,4334 \pm 0,0011 \text{ bar}$

Resultado según MMC obtenido con la aplicación generada a partir de este trabajo:

Mediana: 34,43342 bar; valor más probable: 34,4335 bar; intervalo de confianza para 95,45% de probabilidad = [34,43235; 34,43449]

Según estos resultados:  $U_{MMC}(P) = (34,43449 - 34,43235) / 2 = 0,00107$

Por lo que tomando como resultado para el MMC la mediana del conjunto de resultados, se obtienen resultados idénticos para ambos métodos, observando una diferencia numérica para la incertidumbre de  $0,0011 - 0,00107 = 3 \times 10^{-5}$ .

La tolerancia numérica para el valor de incertidumbre según GUM ( $=0,0011$ ) está dada por:

$$\delta = \frac{1}{2} 10^{-4} = 0,5 \times 10^{-4}$$

Por tanto,  $3 \times 10^{-5} < 0,5 \times 10^{-4}$  y los resultados por ambos métodos pueden considerarse compatibles.

### Ejemplo 3.

#### Estandarización de una solución de Hidróxido de sodio (Ejemplo A2 de EURACHEM / CITAC Guide CG 4)

En este ejemplo se comparan los cálculos detallados en la guía citada con los obtenidos por medio de la aplicación generada en este trabajo por el método de Monte Carlo.

En este caso se estudia la determinación de la concentración de una solución de NaOH por medio de una titulación con biftalato de potasio (KHP). La ec. general del proceso es:

$$C_{NaOH} = \frac{1000 * m_{KHP} * P_{KHP}}{M_{KHP} * V_T}$$

Donde:

$C_{NaOH}$ : concentración de la solución de NaOH [mol/l].

$m_{KHP}$ : masa del KHP (como se pesa por diferencia, la incertidumbre de la pesada se toma por duplicado).

$P_{KHP}$ : pureza del KHP dada como porcentaje en masa.

$M_{KHP}$ : masa molar del KHP [g/mol].

$V_T$ : volumen dispensado en la titulación.

Aquí, según el punto A2.4 step 3 de la guía, la magnitud consta de cuatro componentes: por repetibilidad (considerada en repetibilidad general), de calibración, debida al control de temperatura, y debida al sesgo en la detección del punto final, considerada despreciable en este caso.

Replanteando entonces la ecuación tenemos que:

$$m_{KHP} = 8 * C + 5 * H + 4 * O + K$$

Siendo C, H, O y K el peso atómico de los elementos correspondientes del KHP ( $C_8H_5O_4K$ ).

Por otra parte, como se indica arriba:

$$V_T = \cancel{V_{rep}} + V_{Cal} + V_{Tem} + \cancel{V_{Sesg}}$$

El set de ecuaciones correspondiente a este modelo utilizado en la aplicación es:

$$\begin{aligned} C_{NaOH} &= 1000 * (m_{KHP\_g} - m_{KHP\_t}) * P_{KHP} / (M_{KHP} * V_T) \\ V_T &= V_{Cal} + V_{Tem} \\ M_{KHP} &= 8 * C + 5 * H + 4 * O + K \end{aligned}$$



Datos de entrada:

Magnitud	Valor	Distrib	Param.
Rep	1	Normal	Desvío standar = 0,0005
m_KHP_g	60,5450	Uniforme	Semi intervalo = 1,5E-4
m_KHP_t	60,1562	Uniforme	Semi intervalo = 1,5E-4
P_KHP	1,0000	Uniforme	Semi intervalo = 0,0005
V_cal	18,64	Normal	Desvío standar = 0,012
V_Tem	0	Normal	Desvío standar = 0,006
C	12,0107	Uniforme	Semi intervalo = 0,0008
H	1,00794	Uniforme	Semi intervalo = 0,00007
O	15,9994	Uniforme	Semi intervalo = 0,0003
K	39,0983	Uniforme	Semi intervalo = 0,0001

Resultados según método GUM, indicado en EURACHEM CG 4 A2.6 Step 5:

$$C_{\text{NaOH}} = 0,1021 \pm 0,0002 \text{ mol.l}^{-1}$$

Recalculando y expresando con las cifras significativas correctas según GUM (JCGM 100), quedaría en:  $C_{\text{NaOH}} = 0,10214 \pm 0,00020 \text{ mol.l}^{-1}$

Resultado según MMC obtenido con la aplicación generada a partir de este trabajo:

Mediana: 0,10214 mol.l<sup>-1</sup>; valor más probable: 0,10214 mol.l<sup>-1</sup>; intervalo de confianza para 95,45% de probabilidad = [0,10194; 0,10233].

Según estos resultados:  $U_{\text{MMC}}(C_{\text{NaOH}}) = (0,10233 - 0,10194)/2 = 0,000195 \text{ mol.l}^{-1}$

La tolerancia numérica para el valor de incertidumbre según GUM (=0,00020) está dada por:

$$\delta = \frac{1}{2} 10^{-5} = 0,5 \times 10^{-5}$$

Por lo que el apartamiento obtenido entre las incertidumbres según ambos métodos es menor que la tolerancia numérica, mientras que los resultados obtenidos son idénticos, por lo que ambos métodos pueden considerarse compatibles.

## Validación de resultados

Se evalúa la validez de los resultados obtenidos para el MMC con la aplicación generada, realizando la simulación en lenguaje R (versión 2) con las mismas magnitudes de entrada.

El Gráfico 6 muestra en forma gráfica las diferencias entre los resultados de Mediana y U, obtenidos de simulaciones realizadas con ambas aplicaciones. Se observa que el apartamiento entre los valores de mediana e incertidumbre es menor que la tolerancia numérica de la incertidumbre para todos los casos, por lo que los resultados de ambas aplicaciones se consideran compatibles.

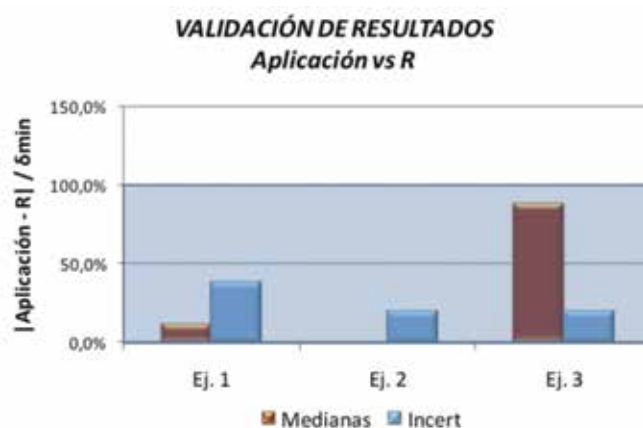


Gráfico 6. Impacto porcentual de la diferencia entre los resultados obtenidos con la aplicación y el software R.

	Ejemplo 1		Ejemplo 2		Ejemplo 3	
	Densidad del aire		Presión generada		Estandarización de NaOH	
Parámetro	Aplicación	Lenguaje R	Aplicación	Lenguaje R	Aplicación	Lenguaje R
Mediana	1,19401	1,194004	34,43342	34,43342	0,10214	0,1021356
Mín (P= 95%)	1,19045	1,190448	34,43235	34,43234	0,10194	0,1019396
Máx (P= 95%)	1,19758	1,197540	34,43449	34,43450	0,10233	0,1023320
U=(Máx – Mín)/2	$3,565 \times 10^{-3}$	$3,546 \times 10^{-3}$	$1,07 \times 10^{-3}$	$1,08 \times 10^{-3}$	$1,95 \times 10^{-4}$	$1,96 \times 10^{-4}$
δ (Tol. num.)	$0,5 \times 10^{-4}$		$0,5 \times 10^{-4}$		$0,5 \times 10^{-5}$	

## Conclusiones

Se indican a continuación las conclusiones inferidas para los distintos aspectos estudiados en la aplicación.

**Nº de repeticiones.** Los resultados permiten concluir que para la aplicación del MMC se requiere realizar un mínimo de 100 mil repeticiones. Por debajo de este valor pueden encontrarse diferencias sensibles en los resultados para las mismas magnitudes de entrada, lo que convierten el método en poco reproducible. Por encima de 100 mil repeticiones tanto los resultados obtenidos para el modelo como sus incertidumbres asociadas serán idénticas, independientemente del número de repeticiones.

**Algoritmos de generación de aleatorios.** Existe homogeneidad razonable en los resultados obtenidos utilizando diferentes algoritmos, por lo que la sustitución de los algoritmos predeterminados por otros de mejor desarrollo no aporta mejoras significativas al método cuando es aplicado a la estimación de incertidumbres.

**Interpretación de resultados.** Como valor del mesurando, en las pruebas se tomó la mediana de la población de resultados, para poder expresar la incertidumbre de la misma forma que se hace según el método GUM. Se observa que la mediana y el valor más probable no coinciden en todos los casos, aunque son similares. En los modelos que la FDP resultante sea menos asimétrica, el promedio, la mediana y el valor más probable estarán más alejados; en estos casos es recomendable estudiar la forma de expresar la incertidumbre y el parámetro estadístico a utilizar como resultado.

Por otra parte, en los tres casos estudiados se obtienen resultados compatibles sin apartamientos significativos entre ambos métodos, por lo que tanto el valor de la magnitud de salida, como su incertidumbre pueden considerarse válidos. En caso de que existieran apartamientos mayores que la tolerancia numérica de la incertidumbre, los valores obtenidos según GUM deberían ser corregidos según el método de Monte Carlo, como se recomienda en JCGM 101:2008.

**Validación de la aplicación generada.** Finalmente, las pruebas comparativas realizadas sobre un software de estadística reconocido (Lenguaje R) permiten concluir que los resultados obtenidos en este trabajo son compatibles con el valor obtenido en plataforma R para los tres ejemplos. Esto representa una forma de validación de la aplicación generada, en el entendido de que deben realizarse pruebas más exhaustivas para considerarla correctamente validada.

**Extensión del trabajo.** A la aplicación generada a partir de este trabajo se le dio el nombre de MCM Alchimia y se encuentra disponible para descarga gratuita en el sitio [www.mcmalchimia.com](http://www.mcmalchimia.com). Se alienta a realizar nuevas pruebas y validaciones con modelos de ensayo más exigentes, y se propone adicionalmente dar disponibilidad a la aplicación para el estudio de modelos de regresión.

- BURKS, A. W.; WARREN, D. W.; WRIGHT, J. B. An analysis of a logical machine using parenthesis-free notation. En: *Mathematical Tables and Other Aids to Computation*. 1954, 46(8):53-57.
- DAGPUNAR, J. *Principles of random variate generation*. Oxford: Clarendon Press, 1988
- EURACHEM; CITAC. *EURACHEM / CITAC Guide CG 4: quantifying uncertainty in analytical measurement*. 3rd ed. London: EURACHEM, [s.d.].
- JCGM 100. *Evaluation of measurement data: guide to the expression of uncertainty in measurement* [En línea]. [s.l.]: JCGM, 2008. [Consulta 2 de mayo de 2013]. Disponible en: [http://www.bipm.org/utills/common/documents/jcgm/JCGM\\_100\\_2008\\_E.pdf](http://www.bipm.org/utills/common/documents/jcgm/JCGM_100_2008_E.pdf)
- JCGM 101. *Evaluation of measurement data. Supplement 1 to the "guide to the expression of uncertainty in measurement". Propagation of distributions using a Monte Carlo method* [En línea]. [s.l.]: JCGM, 2008. [Consulta 2 de mayo de 2013]. Disponible en: [http://www.bipm.org/utills/common/documents/jcgm/JCGM\\_101\\_2008\\_E.pdf](http://www.bipm.org/utills/common/documents/jcgm/JCGM_101_2008_E.pdf)
- KREJCI, Roman. *Another random numbers generator (32-bit)* [En línea]. [s.l.]: [s.n.], 1999. [Consulta 2 de mayo de 2013]. Disponible en: <http://www.rksolution.cz/delphi/tips.htm>
- MARSAGLIA, G.; BRAY, T. A convenient method for generating normal variables. En: *SIAM Rev.* 1964, 6(3):260-264.
- MATSUMOTO, M.; NISHIMURA, T. Mersenne twister: a 623-dimensionally equidistributed uniform pseudorandom number generator. En: *ACM Trans. on Modeling and Computer Simulation*. 1998, 8(1):3-30.
- PICARD, A.; DAVIS, R. S.; GLÄSER, M.; FUJII, K. Revised formula for the density of moist air. En: *Metrologia*, 2008, 45(2):149-155.
- SAUCIER, R. *Computer generation of statistical distributions* [En línea]. [s.l.]: Army Research Laboratory, 2000. [Consulta 2 de mayo de 2013]. Disponible en: <http://ftp.arl.mil/random/random.pdf>
- THOMAS, D.B.; LUK, W.; LEONG, P.H. Gaussian random number generators. En: *ACM Comput. Surv.* 2007, 39(4):Article 11.

## Reconocimientos

Se agradece al Ing. Luis Mussio, primer piloto de pruebas de la aplicación, cuyas acertadas respuestas y cuestionamientos en la etapa de desarrollo permitieron optimizar y dar mayor utilidad a la misma desde sus versiones iniciales.

## Referencias

- BOX, G. E. P.; MULLER, M. E. A note on the generation of random normal deviates. En: *The Annals of Mathematical Statistics*. 1958, 29(2)610:611.