# Optimización de un esquema "Ocuppancy Problem" orientado a E - Voting

**Jeroen van de Graaf [1] ; Germán Montejano [2 3]; Pablo García [3]**

[1] Departamento de Ciência da Computação
Universidade Federal de Minas Gerais
Av. Antonio Carlos, 6627 – 31270-010 - Belo Horizonte – Minas Gerais - Brasil
Tel.:+55-3409-5836
jvdg@dcc.ufmg.br – web: http://www.dcc.ufmg.br/~jvdg

[2] Departamento de Informática
Universidad Nacional de San Luis
Ejército de los Andes 950 – (5700) San Luis – San Luis – Argentina
Tel.: +54-2652-424027 – Int. 251
gmonte@unsl.edu.ar – web: http://www.unsl.edu.ar

[3] Departamento de Matemática
Universidad Nacional de La Pampa
Av. Uruguay 151 – (6300) Santa Rosa – La Pampa – Argentina
Tel.: +54-2954-425166– Int. 125
pablogarcia@exactas.unlpam.edu.ar – web:http:/unlpam.edu.ar

## RESUMEN

Este trabajo tiene el objetivo de presentar un enfoque alternativo al clásico "problema de la ocupación", el cuál consiste en analizar las probabilidades relacionadas con disponer de una serie de bolillas que se colocarán, de manera aleatoria, en un conjunto de ranuras. El problema se generaliza con facilidad y resulta de sumo interés en múltiples aplicaciones prácticas. Desde una implementación informática, se piensa en un vector capaz de almacenar un dato en cada posición.
En particular, si pensamos en esquemas de voto electrónico, muchos modelos actuales basan su nivel de seguridad, en lo referente a la privacidad, en que la elección de la posición exacta en la que un voto se almacenará sea auténticamente aleatoria.
Tal exigencia, sin embargo, genera la imposibilidad de evitar que dos o más votos se alojen en una misma posición, lo cuál deriva en la pérdida de todos los sufragios coincidentes en una posición determinada del vector. Por lo tanto, minimizar la probabilidad de colisiones es un punto fundamental.
La aproximación teórica más conocida se conoce como "Birthday Paradox" y demuestra que para obtener niveles razonables de seguridad, es necesario implementar un vector de un tamaño significativamente mayor que la cantidad de votos que se deba administrar.
El presente trabajo entonces, propone un modelo alternativo, consistente en la aplicación de $n$ vectores paralelos, con $n \geq 2$. Cada sufragio se replicará en todos los vectores, en posiciones aleatorias y potencialmente diferentes en cada caso. En tal esquema, un voto específico sólo se perderá si colisiona en todos los canales paralelos.
Para dejar clara la mejora que el esquema implementa, se realizan comparaciones en las que la cantidad total de slots implementados es la misma, pero distribuidos en dos o más vectores paralelos.
Se presentan fundamentos teóricos existentes y algunos resultados obtenidos con la utilización de un simulador, programado ad hoc, el cuál genera procesos eleccionarios, manejando los parámetros necesarios y entrega resultados sobre el comportamiento de las variables que se desea observar.

## *PALABRAS CLAVE*

***Colisión, slot, seguridad incondicional, anonimato, voto electrónico, Dining Cryptographers, Birthday Paradox, Occupancy Problem.***

## CONTEXTO

Este trabajo se enmarca en un convenio de intercambio entre la Universidade Federal de Minas Gerais (Brasil) y la Universidad Nacional de San Luis (Argentina). El mismo posibilita el intercambio de alumnos y docentes entre ambas instituciones, promoviendo, además, el desarrollo de proyectos comunes. En particular, corresponde a la articulación de la Maestría en Ingenierá de Software (UNSL) y el Mestrado em ciência da Computação (UFMG), ambos acreditados con la máxima categoría en Argentina y Brasil, respectivamente.

## 1. INTRODUCCIÓN

Los esquemas tradicionales de voto electrónico proveen, mayoritariamente, un nivel de seguridad computacional en lo referente al anonimato. Al mismo tiempo, proporcionan seguridad de tipo incondicional, durante el proceso de votación. En [8] se plantea que tal enfoque no resulta apropiado y debería invertirse, dado que la protección del proceso se lleva a cabo durante un tiempo limitado, ya que apenas termina la elección, los resultados se hacen públicos. En cambio, el anonimato debe protegerse indefinidamente. Es facil imaginar las consecuencias de que se conozca la trayectoria, como votante, de un candidato que se postula en un proceso electoral actual.
Un esquema que busca dar respuesta a tal planteo se basa en implementar un vector de slots, capaces de almacenar votos. En los productos de software que adhieran a tal modelo, el anonimato podrá mantenerse de manera incondicional si la posición en la que se almacena cada voto es auténticamente aleatoria. En consecuencia, la probabilidad de que se produzcan colisiones (es decir, que dos o más votos se alojen en el mismo slot) presenta un valor mayor que cero.
Un antecedente teórico relacionado es conocido como *Ocuppancy Problem* y es profusamente abordado en los textos relacionados con la teoría de probabilidades. Por ejemplo, [3] lo describe detalladamente y [2] propone una fórmula matemática de aproximación basada en la distribución de Poison. En cualquier caso, los valores obtenidos muestran que la relación slots – votantes debe mantenerse alta si se desea obtener niveles razonables de seguridad.
Por ejemplo, la muy conocida paradoja de cumpleaños (Birthday Paradox) se refiere al hecho, poco intuitivo, de que dado un grupo de 23 personas, la probabilidad de que dos de ellas cumplan años el mismo día es cercana a 0,5. Esto parece indicar que si se pretenden obtener niveles razonables de seguridad para un esquema de voto electrónico, la cantidad de almacenamiento que deberá implementarse será muy elevado en comparación con el tamaño de la muestra.
En particular, las aplicaciones de voto electrónico exigen altísimos niveles de seguridad. Debe tenerse en cuenta que un sistema de voto electrónico no resultará aceptable si no presenta ventajas significativas con respecto a los esquemas tradicionales, con boletas de papel. Y debe reconocerse que tales esquemas no presentan problemas severos en ese punto en particular.
En [8] se describe un esquema basado en un vector de slots para almacenar votos. Sin embargo, la aplicación de tal solución, tal como se observa en Birthday Paradox, resulta muy desfavorable si se desea generalizar el esquema para su aplicación en E-Voting, dado que el número de slots a implementar para proporcionar los niveles de seguridad requeridos en tal aplicación crece de manera muy significativa. Sobre todo teniendo en cuenta los enormes niveles de exigencia de la aplicación, que define relaciones de poder muy importantes en las sociedades actuales.
Por lo tanto, el presente trabajo propone una variante que optimiza el almacenamiento que se destine a los votos. Consiste en implementar múltiples vectores iguales, de manera tal que la suma de los slots coincida con la del vector único que se implementa en el modelo original. La idea es que cada voto se replique en todos los vectores, pero en posiciones aleatorias potencialmente diferentes en cada caso.
Ante ese planteo, es evidente que un voto específico solamente será perdido si colisiona en todos los vectores implementados. Luego, se desea establecer si la probabilidad de pérdida es mejor en este escenario que en el original. Aparecen dos situaciones contradictorias:

1. El número de colisiones va a aumentar, dado que la totalidad de los sufragios se replicará en cada uno de los vectores implementados.
2. Por propiedad de los sucesos independientes, la probabilidad de que un voto se pierda efectivamente, es el producto de las probabilidades de que colisione en cada uno de ellos.

Se busca demostrar que la segunda propiedad tiene un peso mucho mayor y que, manteniendo ciertos límites en la relación entre la cantidad de sufragios y el tamaño de cada vector individual se obtendrán resultados muy superiores en términos de eficiencia.

## 2. EXPERIMENTOS REALIZADOS

Inicialmente se analiza la situación si se implementa un único vector. Se definen los siguientes parámetros:

*m*: Cantidad de slots que se implementan.
*n*: Cantidad de votantes.

Luego definimos el evento:

X=”se pierde el voto 1”.

La probabilidad de X se define de la siguiente manera:

$$P(X) = 1 - P(\overline{X})$$
$$P(X) = 1 - P(x_1 <> x_j \ \forall j > 1)$$
$$P(X) = 1 - (1/m)\,(1/(m-1))^{n-1}$$

Se busca definir la probabilidad de que no se pierda ningún voto. Obviamente, la probabilidad de que un voto no sea efectivizado en el recuento final es similar para todos los casos.

Se define la siguiente variable aleatoria:

Y=”Cantidad de votos perdidos”.

La variable aleatoria Y responde a una distribución binomial, cuyos parámetros serán (m,P(X)). En consecuencia, la probabilidad de que al menos un voto no sea contabilizado es:

$$P(Y \geq 1) = 1 - P(Y=0)$$
$$P(Y \geq 1) = 1 - (1 - P(X))^{n}$$
$$P(Y \geq 1) = 1 - (1 - 1 - (1/m)(1/(m-1))^{n-1})^{n}$$

Sin embargo, tales desarrollos presentan la dificultad de que, aplicados a valores que puedan resultar de interés para la problemática de voto electrónico, no son calculables, dado que algunos de los valores involucrados son demasiado pequeños como para ser representados en los sistemas tradicionales.

Lo mismo ocurre con una aproximación propuesta en [2]. La aproximación, basada en la distribución de Poison, muestra que para valores grandes de *m* y *n*, la siguiente ecuación es una aproximación aceptable.

$$P(n ; \lambda) = e^{-\lambda} (\lambda^{n} / n!)$$

Tal resultado corresponde a una distribución de Poison, en la que $\lambda$ toma el siguiente valor:

$$\lambda = me^{-n/m}$$

Tal como se describió para fórmulas previas, aparece un inconveniente: no resulta accesible el cálculo del factorial involucrado en la fórmula si se desea aplicar a valores que resulten de interés. Por ejemplo, la versión de LibreOffice 3.5.7.2, en su planilla de cálculo Calc, sólo puede calcular factoriales hasta 170. Cabe mencionar que un recinto de votación electrónica en Brasil es de alrededor de 500 votantes.

Por lo tanto, el enfoque seleccionado para verificar el interés del enfoque consiste en la aplicación de simulaciones. Se implementa un simulador con las siguientes características:

Inputs:

- *m*: cantidad de slots.
- *n*: cantidad de votantes.
- *c*: cantidad de canales paralelos.
- *r*: cantidad de procesos eleccionarios que se simulan en la presente sesión del simulador.

Outputs:

- Votos efectivos acumulados (*vea*): indica la cantidad exacta de votos que no se perdieron, teniendo en cuenta la totalidad de las corridas.
- Votos perdidos acumulados (*vpa*): expresa la cantidad de votos perdidos en total. Obviamente:

  $nr = vea\ vpa$

- Cantidad de corridas con Pérdidas (*cccp*).
- Cantidad de corridas sin pérdidas (*ccsp*). Es evidente que :

  $r = cccp + ccsp$

- Media del generador aleatorio (*mga*): este valor se obtiene a los efectos de verificar la calidad de la muestra generada.
- Porcentaje de Votos Perdidos (*pvp*), referido a la totalidad de simulaciones de una sesión determinada.
- Mejor caso (*mc*): indica cuál es el menor número de votos perdidos en algún acto eleccionario, considerando todas las corridas.
- Peor caso (*pc*): idem al anterior para indicar el mayor número de votos perdidos en alguna corrida.

El programa utiliza un generador aleatorio que responde a una distribución uniforme. Su funcionamiento consiste en generar, para cada voto, una posición al azar para cada uno de los canales paralelos. Luego, al momento de colocar un voto se sigue una de las dos conductas siguientes:

- Si el slot está vacío, se guarda allí un número, correspondiente al número de orden del votante.
- Si el slot está ocupado, se suma a lo que ya contiene, una constante mucho mayor que el número de votantes, lo cuál permite detectar el número exacto de votos que colisionaron en un slot específico. Para todos los experimentos utilizados, dicha constante se fijó en 1.000.000, valor mucho mayor que el número de votantes utilizado (480).

## 3. RESULTADOS OBTENIDOS/ESPERADOS

Los resultados preliminares, basados en simulaciones, muestran una marcada optimización en la utilización del almacenamiento. A los efectos de ilustrar tal situación se muestran algunos resultados obtenidos. Para todos ellos se eligieron dos parámetros fijos:

- Cantidad de votantes: 480.
- Cantidad de corridas: 1.000.000.

En el primer caso, se implementaron 4800 slots y se vigila la variable “cantidad de corridas sin pérdidas”. La figura 1 grafica los resultados obtenidos con la implementación de 1, 2, 4, 5, 6, 8, 10 y 12 canales paralelos. En la misma puede observarse que el valor óptimo se obtiene con la utilización de 6 u 8 canales.

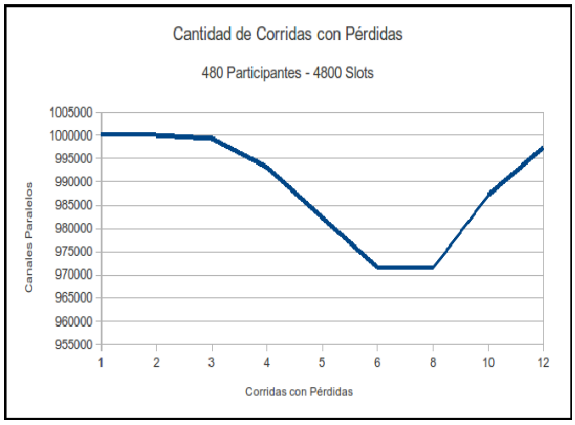


Fig1: Cantidad de Corridas con Pérdidas (4800 slots)

La mejora progresiva que se obtiene aumentando desde 1 a 6 canales es producto de la propiedad mencionada previamente relacionada con la independencia de los sucesos. Tal optimización se pierde cuando se utilizan más de 8 canales porque la cantidad de colisiones en cada canal unitario aumentan significativamente porque el tamaño de cada vector unitario es igual a la cantidad de votantes en el caso de utilizar 10 canales e incluso menor si se usan 12 canales, en cuyo caso cada canal tendrá sólo 400 slots, para 480 votantes.

El segundo ejemplo es similar, pero se implementan 9600 slots. Como muestra la figura 2, la eficiencia crece de manera continua con el agregado de canales. Por ejemplo, con un sólo canal, sobre un millón de corridas, no se perdieron votos solamente en 5. En cambio, con 12 canales, se registraron exactamente 971901 corridas en las que no se perdieron votos. La forma de la curva se modifica con respecto a la figura anterior, porque aún con 12 canales, la medida de cada uno, es siempre mayor que la cantidad de votantes (800 slots para 480 votantes si se usan 12 canales).

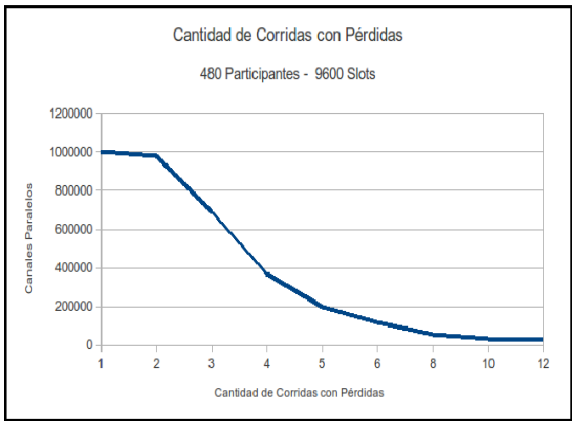


Fig 2: Cantidad de corridas con pérdidas (9600 slots)

Obviamente, los resultados mejoran si se aumenta el número total de slots. Sin embargo, el punto que se desea resaltar es la optimización obtenida, al incorporar más canales paralelos, con un mismo número total de slots.

Otra variable que se ha investigado es el comportamiento del peor caso, es decir, si se realizan 1.000.000 de corridas, cuál es el mayor número de votos que se perdió en alguna de ellas. La figura 3 muestra significativos progresos al implementar vectores paralelos.

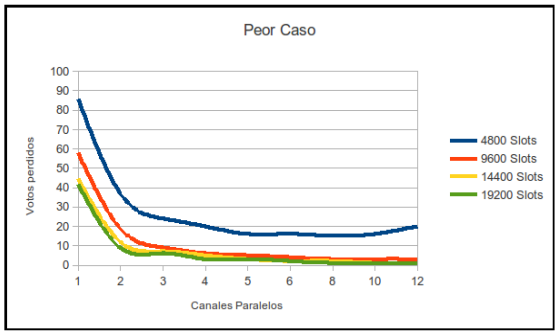


Fig. 3 – Evolución del peor caso

Otro valor que demuestra el nivel de optimización que proporciona el nuevo esquema, pasa por el total de datos efectivos acumulados tras 1.000.000 de simulaciones. Este magnitud implica cuantos votos se perdieron en total. En una simulación con 12 canales paralelos, 480 participantes y 19200 slots sólo se perdieron 38 votos, totalizándose 479.999.962 votos efectivos. Dependiendo de los valores de seguridad requeridos, los valores observados para 9600 y 14400 slots también podrían ser aceptables. La situación se ilustra en la figura 4.

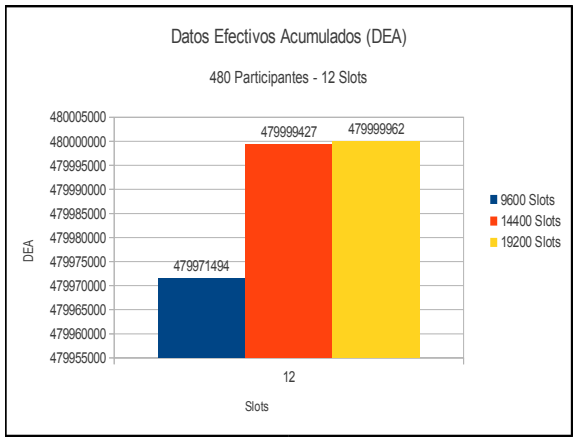


Fig. 4 – Datos efectivos acumulados

En definitiva, los resultados obtenidos muestran una mejora sustancial en el manejo de las colisiones, utilizando la misma cantidad total de slots.

Con respecto a los pasos futuros, mientras se está desarrollando este documento, se continúa investigando con el objetivo de obtener conclusiones matemáticas que permitan cuantificar de manera precisa el nivel en que mejora el esquema con la incorporación de vectores paralelos. Si bien las expericencias realizadas hasta el momento permiten predecir una optimización sustancial en la utilización del almacenamiento, se busca definir de manera formal cuáles son los niveles exactos de optimización que el enfoque aporta.

En definitiva, se desea, por ejemplo, responder de manera matemática formal preguntas como las siguientes:

- ¿Cuál es el número óptimo de vectores para una cantidad de votantes específico para una cantidad determinada de slots?
- ¿Cuál es el número mínimo de slots que debe implementarse y cómo deben organizarse los mismos para que la probabilidad de perder al menos un voto sea menor que un valor deseado?

También se desea obtener conclusiones definitivas relacionadas con posibles herramientas de recuperación post colisión. Por ejemplo, si en cada réplica de un voto determinado se coloca información relacionada con las ubicaciones del mismo voto en los demás slots paralelos, se puede aplicar una simple operación XOR para recuperar votos que hayan colisionado en todas sus instancias, siempre que al menos una de tales colisiones sea simple y con otro sufragio del cuál no se hayan perdido todas las instancias.
Se espera obtener conclusiones formales de todos los tópicos mencionados.

## 4. FORMACIÓN DE RECURSOS HUMANOS

El tema presentado está siendo desarrollado con mayor nivel de detalle en la tesis de maestría de uno de los autores (García), en la cuál los otros dos (van de Graaf y Montejano) son director y codirector, respectivamente. La misma se titula “Optimización de un Esquema de Dinig Cryptographers Asincrónico” y sera defendida durante el presente año en el ámbito de la Universidad Nacional de San Luis (UNSL) para obtener el título de Magister en Ingeniería de Software. Dicha maestría está calificada como “A” en el ámbito de la CONEAU. Además de eso, García realizó un intercambio durante todo 2012 en el ámbito de la Universidade Federal de Minas Gerais (UFMG), en Belo Horizonte, Brasil, a través de una beca CAPES - CAFPBA, obteniendo allí los créditos necesarios para una doble titulación, por lo cuál dicha institución otorgará el título de Mestre em ciência da Computação, carrera de posgrado que también tiene la máxima calificación en el ámbito de la educación superior brasileña. Se espera mantener y profundizar la relación entre ambas instituciones.
También se busca derivar otras tesis de maestría que presenten de manera rigurosa esquemas que apliquen estas técnicas en un esquema concreto de voto electrónico.
La tesis mencionada abarca otra problemática que deja abierta una línea de investigación: la obtención de métodos probadamente eficientes en esquemas que aseguren la seguridad incondicional del anonimato de quienes intercambian mensajes en una red. Ese punto debería derivar en una tesis doctoral que aborde de manera rigurosa enfoques alternativos en ese sentido.

## 5. BIBLIOGRAFÍA

1. Chaum D.: '”The Dining Cryptographers Problem: Unconditional Sender and Recipient Untraceability”'. Journal of Criptology. 1988.

2. Feller W.: "An Introduction to Probability Theory and its Applications". Volúmen I. Tercera Edición. John Wiley and Sons. New York, 1957.

3. Flajolet P., Gardy D., Thimonier L.: 'Birthday paradox, coupon collectors, caching algorithms and self-organizing search'. Discrete Applied Mathematics 39, ps. 207-223. North-Holland. 1992

4. Fuster Sabater A., De La Guía Martínez, D., Hernández Encinas L., Montoya Vitini F., Muñoz Masqué J.: "Técnicas Criptográficas de Protección de Datos". 3a Edición actualizada. ISBN: 978-84-7897-594-5. Editorial Ra - Ma. 2004.

5. Mao W.: "Modern Cryptography: Theory and Practice". Prentice Hall -ISBN: 978-0132887410. 2003.

6. Meyer P: "Probabilidad y Aplicaciones Estadísticas". Addison Wesley Iberoamericana. Segunda edición.

7. Trappe W., Washington L.: "Introduction to Criptography with Coding Theory". Prentice Hall. ISBN: 0-13-061814-4. 2002.

8. Van de Graaf J.: "Anonymous One Time Broadcast Using Non Interactive Dining Cryptographer Nets with Applications to Voting". Publicado en: "Towards Trustworthy Elections". Ps 231-241. Springer-Verlag Berlin, Heidelberg. ISBN:978-3-642-12979-7. 2010.