Taller de Matemática Computacional

**AÑO 2017**

**Sebastián A. Bellido**

**DNI: 27.388.144**

Trabajo Práctico Especial

TUDAI – Facultad de Ciencias Exactas - UNCPBA

INTRODUCCIÓN:

La cátedra TMC propone un trabajo práctico especial utilizando la herramienta Octave. Que el alumno desarrolle un motor de Montecarlo mediante llamadas a la función “my\_mex\_service(dni)”, que recibe como parámetro un número entero representando el número de documento del alumno. La misma devuelve un 1 en caso de autorizar el disparo y un 0 en caso de no autorizarlo.

La nave espacial con la que estamos trabajando utiliza un sistema de autenticación bastante obsoleto, que autoriza a realizar disparos láser de acuerdo al DNI introducido.

DESARROLLO

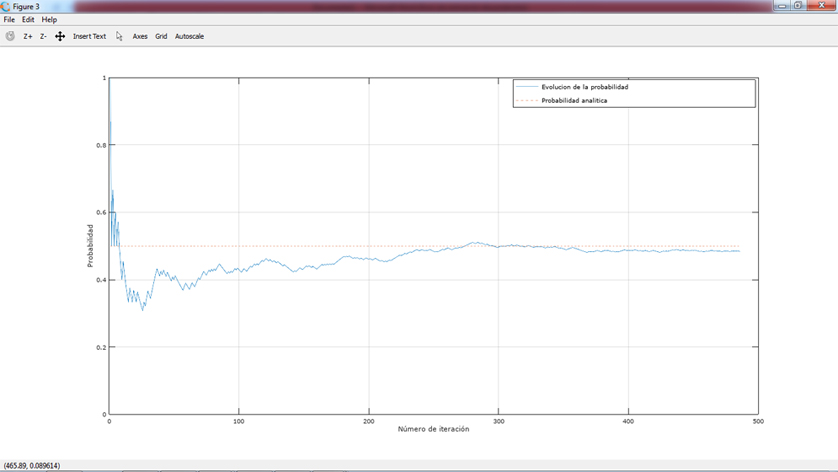
Para llevar a cabo el desarrollo del programa se declaró una función llamada “calcular\_probabilidad\_del\_exito(épsilon, dni)” que va a calcular la probabilidad de obtener dos intentos fallidos, o sea dos negaciones seguidas al preguntar si la nave me permite o no realizar un disparo.

Para esto se declararon las constante “épsilon”, “epsilon1”, “epsilon2” y dni; cada una con un valor predefinido. Éstas servirán como parámetro a la función “calcular\_probabilidad\_del\_exito()”, para obtener los valores deseados.

También fue necesario igualar las variables “intento1” e “intento2” a la función dada “my\_mex\_service(dni)” para poder realizar una sentencia “if” con una operación lógica que compara las variables con el valor booleano 0 (cero). Si se cumple esta sentencia, entonces se incrementa en 1 (uno) nuestra variable “casos\_favorables”.

Basándonos en la regla de Laplace, el programa calcula la probabilidad actual al dividir los casos favorables, o sea las negaciones de disparo, por el total de pruebas realizadas. Este programa además medirá el tiempo en segundos que tardó en realizar todos los cálculos para luego mostrarlos en pantalla.

Finalmente al ejecutar el archivo script\_trabajo\_especial en la ventana de comando de Octave, el programa obtiene los resultados e imprime todos los datos deseados.

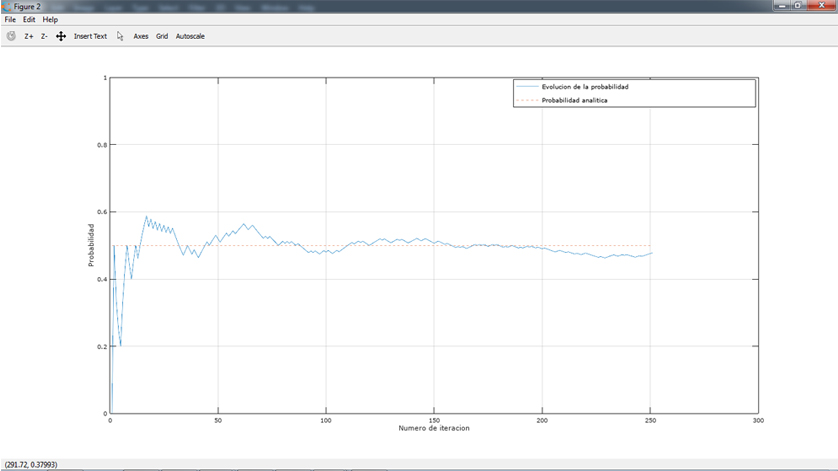


La probabilidad de dos fallos seguidos con un épsilon de 0.1 es: 0.446215

El desvío estándar de las primeras 20 iteraciones es: 0.153388

El desvío estándar de las últimas 20 iteraciones es: 0.00479139

El tiempo en segundos del experimento es: 0.468027

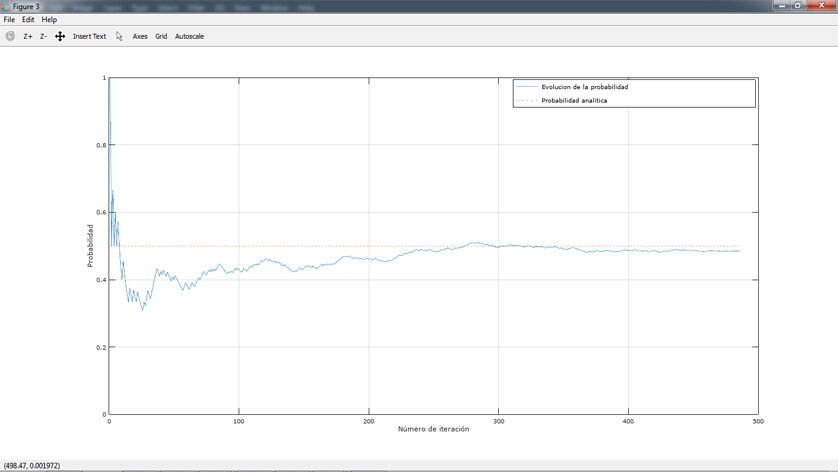


La probabilidad de dos fallos seguidos con un épsilon de 0.01 es: 0.478088

El desvío estándar de las primeras 20 iteraciones es: 0.148065

El desvío estándar de las últimas 20 iteraciones es: 0.00312795

El tiempo en segundos del experimento es: 0.336019



La probabilidad de dos fallos seguidos con un épsilon de 0.001 es: 0.483539

El desvío estándar de las primeras 20 iteraciones es: 0.156243

El desvío estándar de las últimas 20 iteraciones es: 0.000790861

El tiempo en segundos del experimento es: 0.562632

CONCLUSIÓN

La probabilidad de dos fallos seguidos es más acertada al ser de menor valor la constante épsilon, o sea, cuanto más bajo es el valor de épsilon la probabilidad es más estable.

Que desvío estándar varíe sus valores en las últimas 20 iteraciones más que en la primeras 20 iteraciones, tiene que ver con que cuanto más pruebas hago más estable es la probabilidad. (Motor de Montecarlo)

Observando el tiempo que lleva realizar este cálculo al programa, podemos notar que tarda más en converger al ser más pequeño el valor de épsilon, aunque la primera tarda más que la segunda siendo más grande épsilon. Esto tiene que ver con que el programa se inicializa y esto hace que tarde unos milisegundos más.