

# Modelación del proceso de guerra de guerrillas

Juan Sebastián Flórez Jiménez<sup>1, a)</sup> and Francisco Javier Leon Ascencio<sup>1, b)</sup>  
*Departamento de Física, Universidad Nacional de Colombia, Bogotá, Colombia*

## Abstract

In this paper we present the results obtained from the numerical simulation of an algorithm proposed in the article "Universal patterns underlying ongoing wars and terrorism", that describes the behavior of insurgent groups through time in countries over an internal warfare. We observed and tested the power relation between the numbers of attack units with attack strength of the insurgent group and the attack strength. Also when the system reaches a steady-state the exponent of the power-law tends to  $2,69 \pm 0,07$ . The programming language used in the implementation of the algorithm was C++.

## Resumen

En este escrito presentamos los resultados obtenidos de la simulación numérica de un algoritmo propuesto en el artículo "Universal patterns underlying ongoing wars and terrorism", el cual describe el comportamiento de grupos insurgentes a través del tiempo en países sometidos en un conflicto interno. Se observó y se comprobó la ley de potencias entre el número de unidades de ataque en función de la fuerza de ataque de un grupo insurgente, se obtuvo que el exponente límite para la relación número de unidades de ataque con fuerza de ataque vs fuerza de ataque es  $2,69 \pm 0,07$ .

Palabras clave: unidades de ataque, fuerza de ataque

## I. INTRODUCCIÓN

El propósito de este proyecto es implementar el algoritmo propuesto en las páginas 17-18 del artículo "Universal patterns underlying ongoing wars and terrorism"<sup>1</sup>, y reproducir numéricamente las simulaciones propuestas en este artículo. En el artículo mencionado se analiza la evolución temporal de varias guerras, enfatizando en los eventos que producen daños materiales o humanos y la probabilidad que tal evento suceda.

Se presenta un modelo numérico con tiempo discreto para la evolución de los ejércitos involucrados, el cual posee un parámetro que representa la probabilidad de que un frente de cada ejército se divida o se fusione con otro frente; tal parámetro determina el estado estacionario del sistema i.e. la distribución final de la fuerza de ataque del ejército involucrado.

Se simuló el proceso descrito en el artículo mencionado anteriormente y se determinó el tiempo de relajación para varios valores de la probabilidad de dividir un frente del ejército ( $\eta$ ). Se realizaron 5 simulaciones para cada valor del parámetro  $\eta$  con diferentes semillas para el generador aleatorio utilizado, y se graficó la pendiente promedio en función del parámetro  $\eta$ .

El algoritmo fue escrito en C++ y se encuentra en el primer apéndice, se utilizaron las librerías `linear.h` y `Random64.h` que no se encuentran disponibles dentro del paquete básico de c++, pueden ser descargadas del siguiente repositorio de git, al igual que el código utilizado para realizar la simulación:

<https://github.com/JuanSebastianFlorez/ProyectoIntermedioHerramientas>

## II. MÉTODO Y RESULTADOS

El algoritmo implementado se expone a continuación, para futuras referencias se encuentra en la página 18 del artículo "Universal patterns underlying ongoing wars and terrorism"<sup>1</sup>:

Se escoge una unidad con fuerza de ataque  $i$  al azar, de tal forma que la probabilidad de ser escogida sea proporcional a su fuerza de ataque  $s_i$ . Con una probabilidad  $\eta$  la unidad se va a dividir en  $i$  unidades con fuerza de ataque 1. Con probabilidad de  $(1 - \eta)$  la unidad se va a fusionar con otra unidad, que es escogida entre las unidades restantes de forma similar a cómo se escogió la primera, de tal forma que la probabilidad de escogerla es proporcional a su fuerza de ataque. Para realizar las simulaciones con  $\eta > 0$  se tomó como condición inicial la distribución uniforme de la fuerza de ataque en 100000 grupos con fuerza de ataque igual a 1.

### A. Casos extremos

Se tienen dos casos extremos, cuando  $\eta = 0$  y  $\eta = 1$ , que corresponden a la concentración de toda la fuerza de ataque en un solo ejército, y la distribución de la fuerza de ataque en unidades individuales. Tales casos fueron simulados como casos de prueba para el programa creado:

- $\eta = 0$   
Teniendo 100000 unidades de fuerza distribuidas inicialmente de 100000 unidades básicas, se simula el proceso y se obtiene que tras 100000 unidades de tiempo se forma un solo grupo con fuerza de ataque 100000.
- $\eta = 1$   
Teniendo 100000 unidades de fuerza concentradas ini-

<sup>a)</sup>jsflorezj@unal.edu.co, c.c 1032469056

<sup>b)</sup>fjleona@unal.edu.co

cialmente en un solo frente, se simula el proceso anteriormente descrito y se llega al estado estacionario (100000 unidades con fuerza de ataque 1) tras 20 iteraciones en promedio.

### B. Ajuste lineal de los datos

Se obtiene el histograma de frecuencias para  $t=1000000$ , donde se ha corroborado que el sistema alcanza su estado estacionario. Para el ajuste lineal se tiene solamente en cuenta aquellos grupos con frecuencia mayor a 10, esto permite eliminar los datos de los grupos de mayor tamaño, los cuales se alejan de la ley de potencias debido al tamaño finito de la cantidad de la fuerza de ataque de los ejércitos.

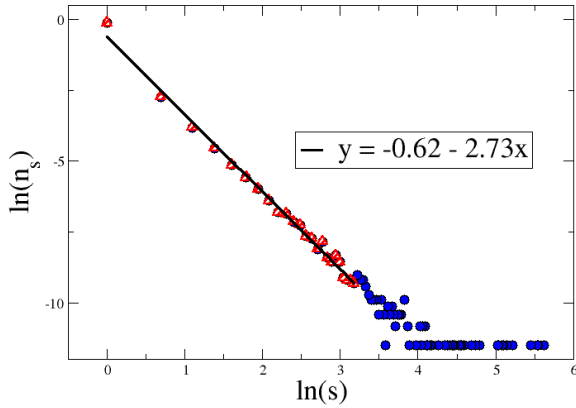


Figura 1. Los datos en azul representan el histograma completo obtenido a partir de la simulación, los datos en rojo representan a los grupo con frecuencia mayor a 10 en la distribución final; el ajuste lineal se realizó con los datos en rojo en todas las simulaciones.  $s$  es el poder de ataque de la unidad considerada, y  $n_s$  es la frecuencia con que aparece una unidad con ese poder de ataque dividido sobre el poder de ataque total.

El código que se encuentra en el repositorio de git mencionado en la introducción tiene funciones para obtener el histograma de los datos, hacer la conversión del histograma a log-log y ajustar una línea a estos datos.

### C. Elección de estado estacionario

Cada simulación se realiza con 1000000 de pasos de tiempo y se grafica la pendiente de ajuste lineal de los datos  $\log(frecuencia_s/100000)$  vs  $\log(s)$ , esto permite determinar el tiempo de relajación del sistema, siendo en todos los casos mayor a 200000 pasos de tiempo. La siguiente gráfica muestra la pendiente del ajuste en función de tiempo para una de las simulaciones realizadas:

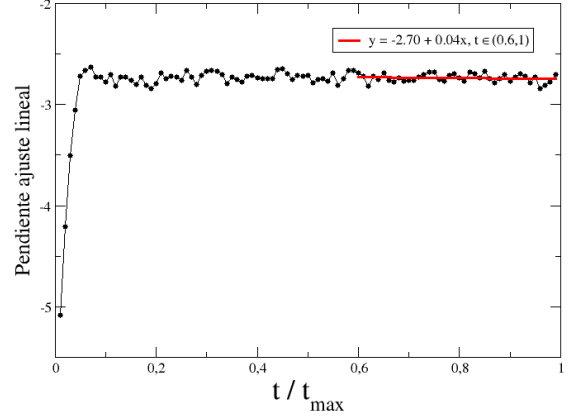


Figura 2. Con  $t/t_{max} = 0,2$  i.e.  $t=200000$  el estado estacionario ya es alcanzado. Se realiza un ajuste lineal de los datos para  $0,6 < t/t_{max} < 1$ , la pendiente de este ajuste es cercana a cero (0,04), luego, el estado estacionario se ha alcanzado. El anterior procedimiento se realiza 5 veces para cada valor de  $\eta$  con semillas diferentes del generador aleatorio. La pendiente registrada es el promedio de tales ajustes en  $t/t_{max} = 0,8$ , y el error en el valor de la pendiente se obtiene de la desviación estándar de los 5 valores obtenidos para cada simulación.

El tiempo de ejecución del programa con  $t=1000000$  y  $N=100000$  es de 8 minutos 36 segundos en promedio. Para hacer el ajuste de las pendientes en función del tiempo y encontrar la pendiente promedio se utilizó Grace.

### D. Exponente en función de $\eta$

Se realizó el ajuste del exponente promedio obtenido para la ley de potencias en función de  $\eta$  (la probabilidad de que el grupo escogido se divida). Se obtuvo la siguiente gráfica, y se puede observar que el ajuste cuadrático es bastante bueno, dando un coeficiente de ajuste mayor a 0.995 i.e.  $R^2 = 9,999$ .

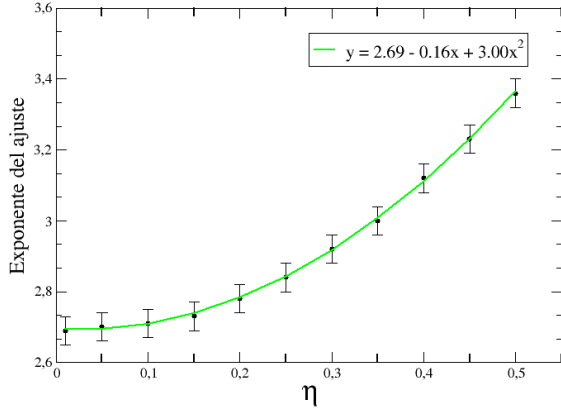


Figura 3. Se aplicó un ajuste cuadrático para el exponente el función de  $\eta$  con  $x \in (0, 0.5)$ , el cual representa una buena aproximación en el rango observado i.e.  $R^2 = 0.999$ .

### III. CONCLUSIONES

- Para  $\eta \lim 0$  se obtiene un exponente límite de 2.69, el cual difiere en un 7,6 % del valor teórico 2.5.
- Para  $x \in (0, 0.5)$  se ajusta el exponente en función de  $\eta$  con una cuadrática, obteniendo un ajuste con  $R^2 = 0.999$ ; luego  $exp = 2.69 - 0.16\eta + 3.00\eta^2$ .
- Debido al limitado poder de cálculo de la computadora utilizada para hacer las simulaciones se tuvo que utilizar un número pequeño para la fuerza total de ataque ( $N=100000$ ) y se simuló 5 veces cada caso con diferentes semillas para el generador aleatorio.
- Los datos se desvían de la ley de potencias para los grupos de mayor tamaño, luego, fue necesario obviar los datos con frecuencia menor a  $10^{-4}$ , obteniendo un coeficiente del ajuste mayor a 0.99 en todos los ajustes realizados.

### REFERENCIAS

- <sup>1</sup> Johnson N. , Spatagan M. , *Universal patterns underlying ongoing wars and terrorism*,
- <sup>2</sup>N. Johnson, M. Spagat, From old wars to new wars and global terrorism