



GRADO EN INGENIERÍA INFORMÁTICA

PROGRAMACIÓN CONCURRENTE Y DE  
TIEMPO REAL

ANÁLISIS

## Práctica 5

**Autor:**

Raúl Arcos Herrera

## Índice

1. primosParalelos.java	2
2. volcadoRed.java	3

## 1. primosParalelos.java

Este problema es de tipo computación numérica, lo que quiere decir que su coeficiente de bloqueo = 0.

A la hora de saber el número de hebras a utilizar, según la ecuación *Subramanian* es el número de hebras del que disponga el sistema, por lo que la solución es utilizar la función *Runtime.getRuntime().availableProcessors()*; para obtener el número de hilos disponibles en el sistema.

## 2. volcadoRed.java

En este caso el problema es de otra naturaleza, debido a su alta interacción de red. La solución es aplicar la ecuación *Subramanian*, debido a que no sabemos el Cb óptimo para este programa se realizarán pruebas con varios valores, una vez encontrado el Cb óptimo (El que mejor rendimiento aporte) utilizamos la ecuación para determinar el número de hebras a utilizar.

Según el análisis empírico del problema, llegamos a la siguiente tabla:

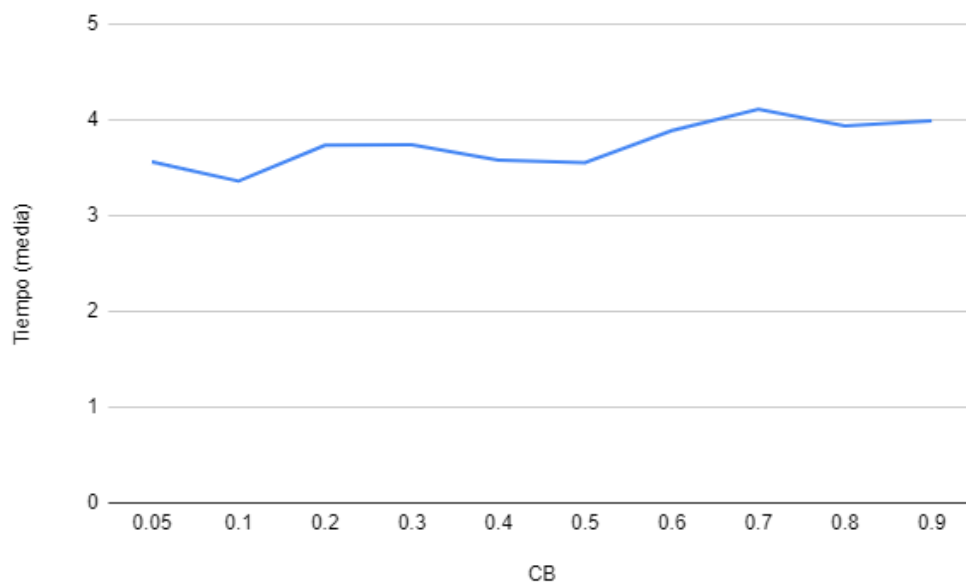


Figura 1: Tiempo frente a Cb.

A pesar de que los resultados no sean muy claros, podemos ver que los resultados óptimos están alrededor de 0.5 Cb, que según la ecuación nos deja con el uso de 24 tareas en la máquina utilizada (12 hilos).