

Practica #6

Sistema Multiagente

20 de marzo del 2018

Introducción

Un sistema multiagente es un sistema computacional integrado por múltiples agentes inteligentes que interactúan entre ellos. Un sistema multiagente es comúnmente utilizado para resolver problemas de gran dificultad para un sistema monolítico.

En cierto modo, un sistema multiagente es un sistema distribuido en un entorno donde la conducta combinada de sistemas de inteligencia artificial realiza acciones en el entorno y a partir del estado del entorno el agente realiza alguna acción. Este tipo de sistemas son descentralizados, es decir, los agentes no siguen ordenes de una unidad central, sino que cada agente es autónomo.

En la realización de la practica 6 se llevo a cabo una simulación de una epidemia de una cantidad de agentes n , donde el agente podría cumplir alguna de las siguientes condiciones:

Sano: El agente se encuentra sano si no se ha contagiado al estar cerca de algún agente infectado

Infectado: El agente estaba infectado, pero tiene una probabilidad de recuperarse y volverse inmune

Inmune: El agente había estado infectado y se volvió inmune.

Para definir los cambios de posición, cada uno de los agentes cuenta con una velocidad horizontal y vertical (X , Y), en cada paso de la simulación los agentes avanzan en función de su posición inicial y su velocidad, dentro de una dimensión dada por $L \times L$, en forma de torus.

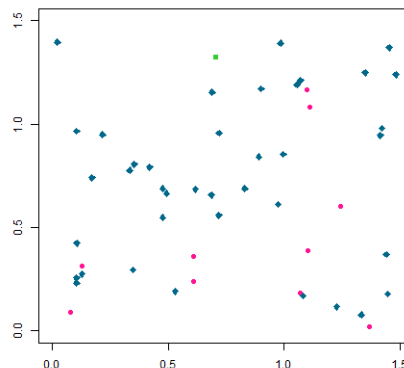


Ilustración 1 Sistema Multiagente

Objetivo

Identificar e implementar las formas mas eficientes de paralelizar y optimizar el código proporcionado en la pagina para la realización de la práctica.

El paralelizado se realizó mediante la creación de clústeres con el paquete doParallel los cuales después ejecuto mediante una función la misma se repitió por el número de agentes iniciales.

Simulación

Para el experimento se tomó como base el código de la página del presente curso donde cada uno de los agentes cuenta con los siguientes parámetros:

Probabilidad de recuperación $p_r = 0.02$

Umbral de infección $r = 0.1$

Dimensión $l = 1.5$

Cantidad de agentes $n=50$

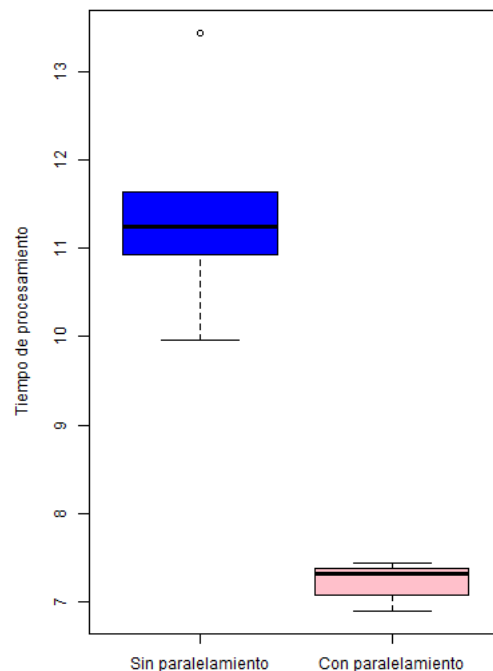


Ilustración 1 Grafica Caja-Bigote de tiempos de ejecución

En la ilustración 2 muestra alrededor del doble de tiempo de ejecución entre un código paralelizado y uno sin paralelizar en este caso el código original, otro dato importante que se aprecia en la ilustración 2 es la distribución del tiempo que es mas amplia en la ejecución del código original.

Reto 1

Se modifico el código nuevamente para verificar el cambio de la epidemia si se le aplicase una vacuna al inicio de la simulación, en la cual aparecen agentes inmunes con una probabilidad fija $p_v = 0.25$

Para comparar la cantidad de infectados en un tiempo determinado con y sin vacuna se realizaron graficas puntuales para una mayor apreciación del máximo de infectados.

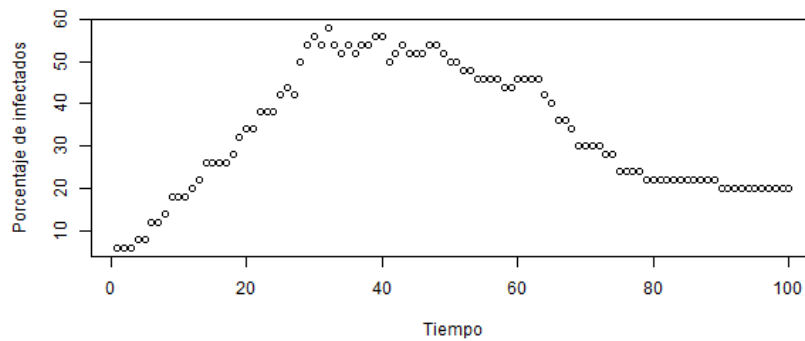


Ilustración 2 a.- Bajo porcentaje de agentes infectados iniciales

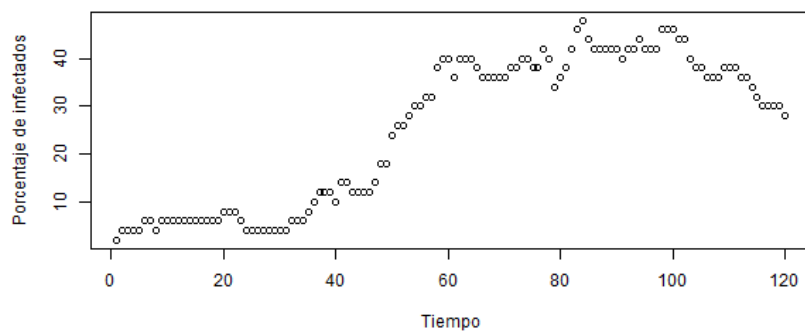


Ilustración 3 b.- Alto porcentaje de agentes infectados iniciales

Como se muestran en la figura 3 a muestra una tendencia mas clara a decaer esto se debe a que se vario la probabilidad de agentes infectados con una alta y baja probabilidad o sea que a medida que vamos aumentando la probabilidad de agentes infectados iniciales es mas lento el proceso a que se vuelvan agentes inmunes.

Conclusiones

Hay una reducción significativa de tiempo de ejecución al paralelizar, alrededor del 50% de tiempo en comparación con el código original.

Si se administra una vacuna antes del inicio de la infección el número máximo de infectados disminuirá, una vez alcanzado el máximo de infectados, decaerá rápidamente gracias a la vacuna.