

Informe sobre el modelo de Hindmarsh-Rose

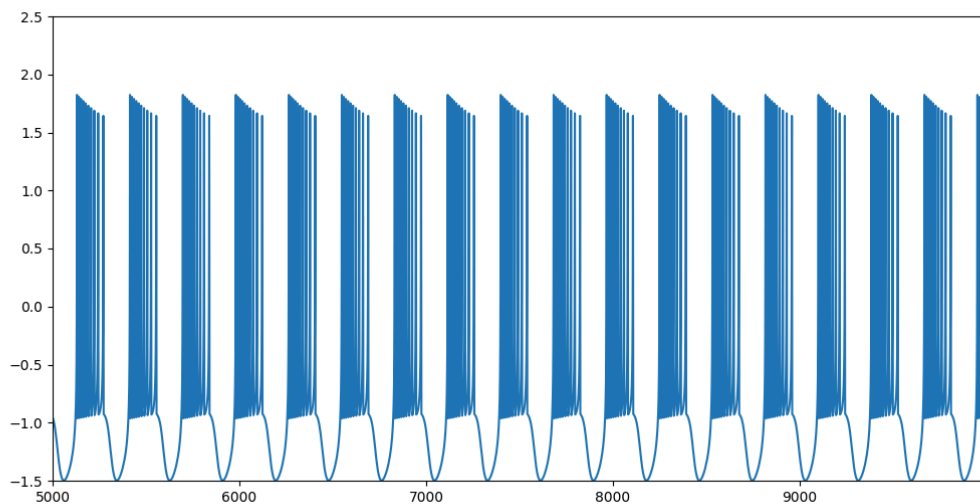
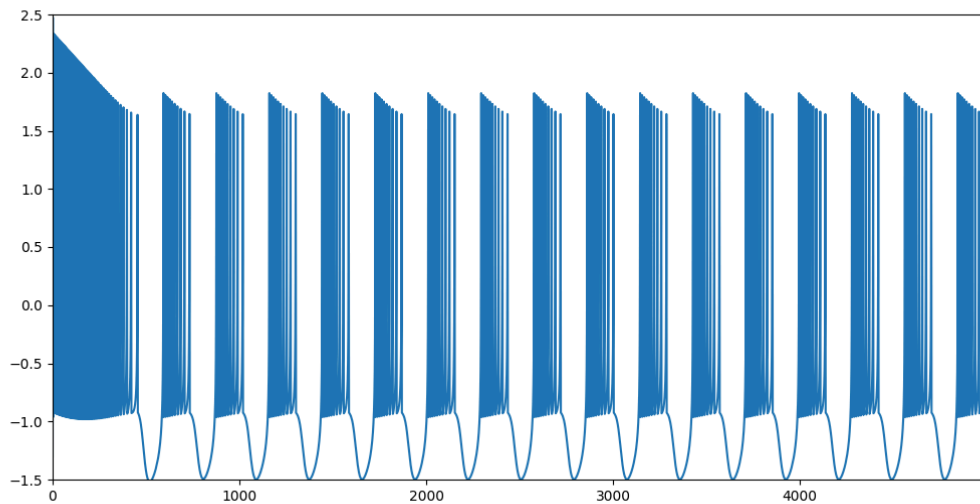
Sergio Hidalgo Gamborino

1.El modelo de Hindmarsh-Rose.....	3
2.Lenguajes usados.....	5
2.1 C++.....	6
2.2 C.....	6
2.2 C/C++.....	6
3.Comparaciones.....	6
3.Conclusión.....	9
Anexos.....	9

1.El modelo de Hindmarsh-Rose

Es un modelo con tres ecuaciones diferenciales que “imita” el comportamiento de una neurona. Este asimismo tiene dos modos, un modo regular, en el que dispara de forma rítmica y otro caótico, donde no parece seguir una serie a priori.

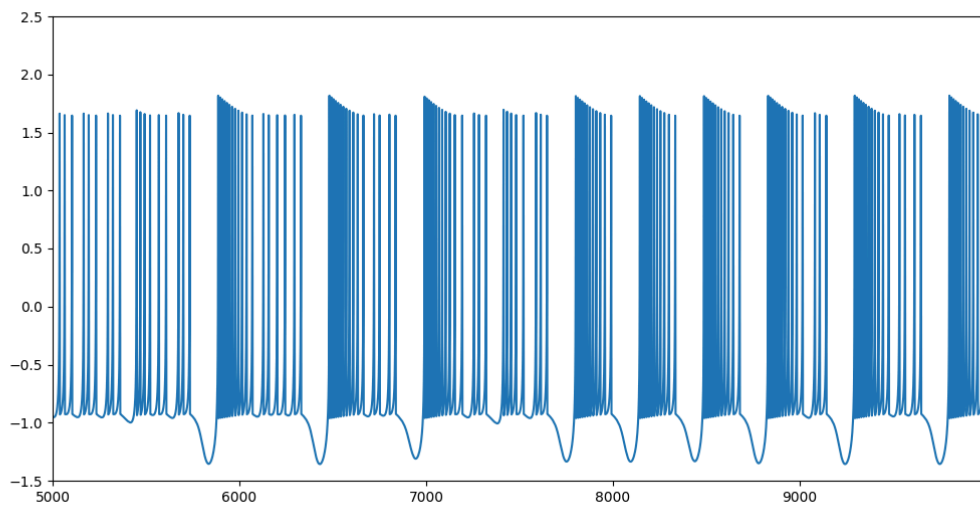
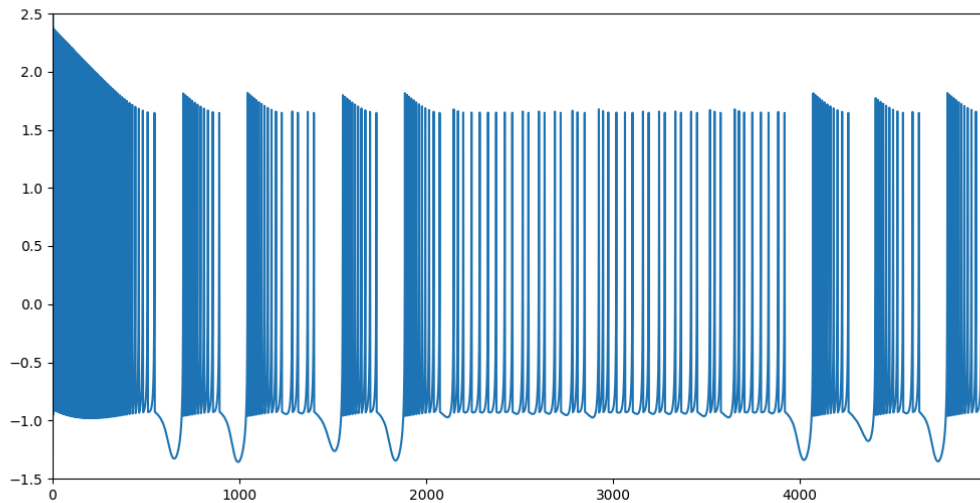
Modo Regular



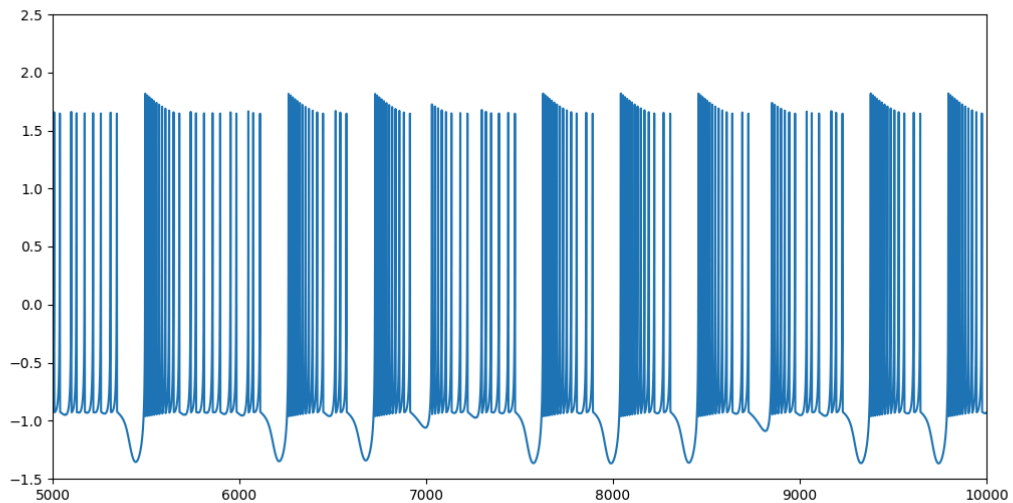
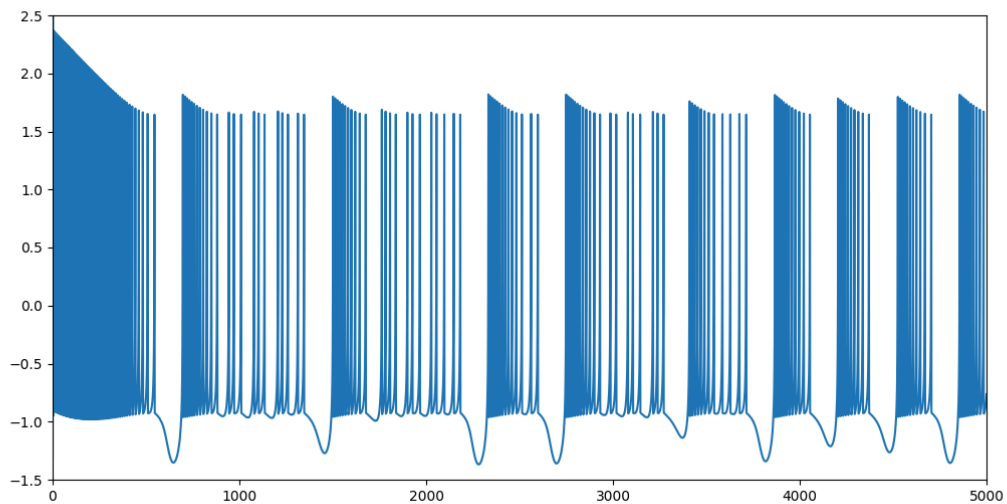
Modo Caótico

El modelo caótico difiere entre la implementación en c y la implementación en c++, esto es probablemente a que en c++ se han usado tipo float para guardar los datos mientras que en c se han usado el tipo double.

En c:



En c++:



2.Lenguajes usados

Los lenguajes empleados para la programación del modelo han sido C/C++. Se han probado 3 combinaciones distintas:

1. C++ usando sus herramientas al completo.
2. C estricto, sin ni tan siquiera definición de tipo.
3. El código de C pero compilado en C++.

Como el programa necesita ser hard real-time, se ha hecho un pequeño estudio de rendimiento a las 3 implementaciones para tener una la idea de por dónde deberá ir la implementación final.

2.1 C++

La primera implementación (archivos : “Model.h”, “Model.cpp”, “HindmarshRose.h”, “HindmarshRose.cpp” y “main_cpp.cpp”) tiene como problema de rendimiento tanto la escritura de los datos, como la ejecución del programa (aunque esta última se ha podido paliar sustituyendo los vectores por arrays).

2.2 C

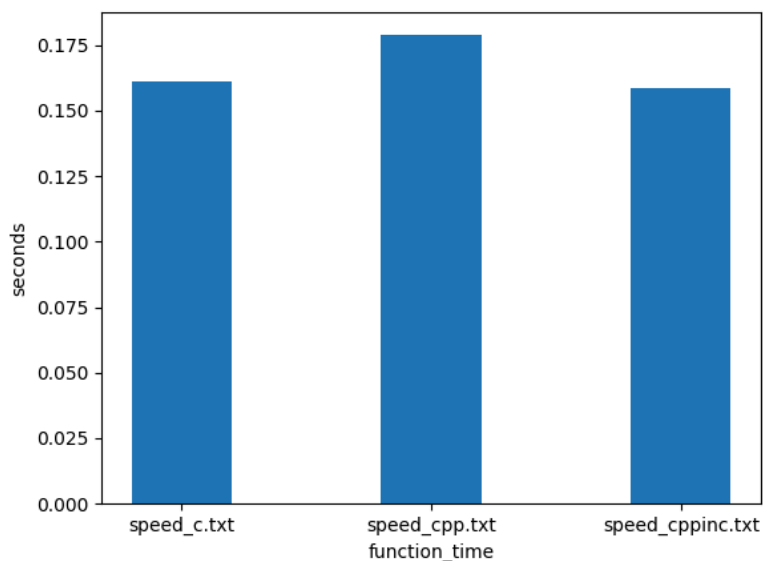
Esta implementación (archivos : “hindmarsh_rose.h”, “hindmarsh_rose.c”, y “main_c.c”) en C es la ganadora respecto al rendimiento general, pero cuenta con la desventaja de que dicho lenguaje se ha quedado atrás respecto a los lenguajes más modernos, además, los decimales pierden algo de precisión y se debe usar la función “ceil” de la biblioteca “math.h” para reservar la memoria correctamente.

2.2 C/C++

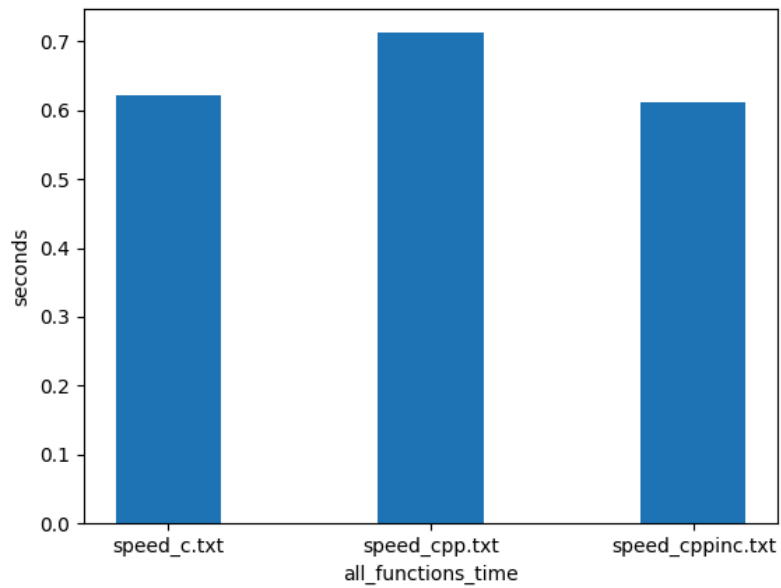
Escrito en C pero compilado en c++ (archivos : “hindmarsh_rose.h”, “hindmarsh_rose_cppinc.cpp”, y “main_cppinc.cpp”), al hacer uso de características en c que no están optimizadas para el compilador de c++, como en la interacción con ficheros.

3.Comparaciones

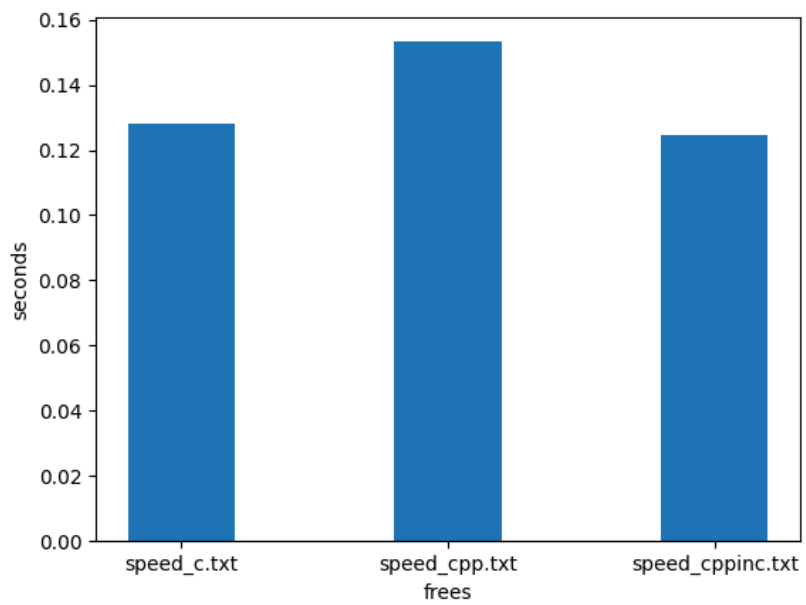
Tiempo de ejecución de una función



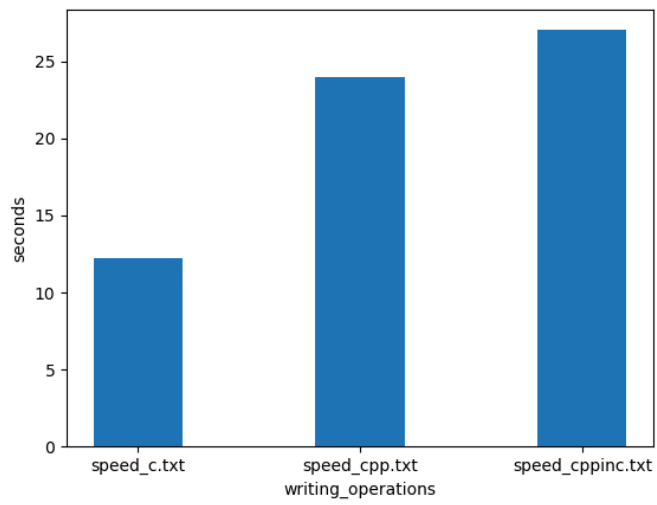
Tiempo de las 4 funciones principales



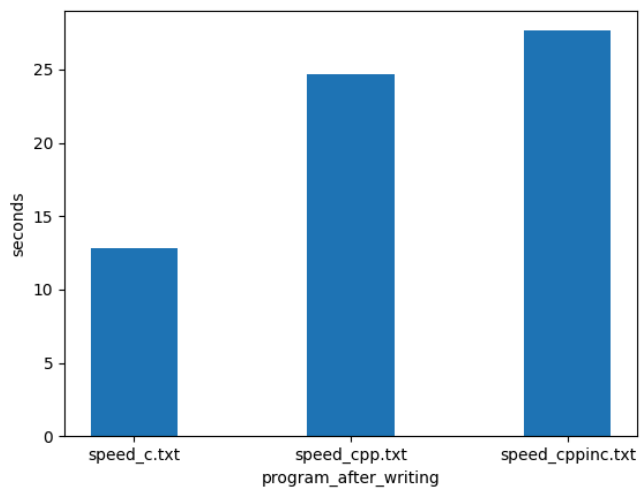
Tiempo de liberación de memoria



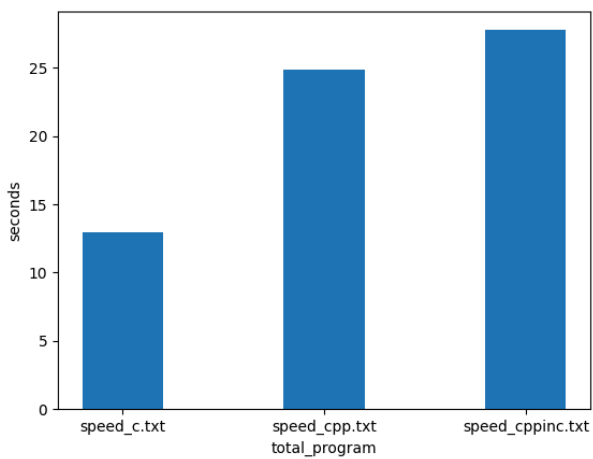
Tiempo del programa en escritura



Tiempo del programa tras la escritura



Tiempo total



3.Conclusión

C es bastante más rápido que c++, respecto a la escritura de datos, aunque respecto a la ejecución de las funciones no hay una gran diferencia. Sería recomendable aprovechar c++ para hacer las clases y los cálculos y probar si se puede hacer un wrapper con una función escrita en c (aunque habría que probar si realmente merece la pena). Además hay que tener en cuenta que no se han aplicado flags de optimización.

Anexos

- Las gráficas del modelo se encuentran en "graphs/HR/"
- Las gráficas de velocidad se encuentran en "graphs/timing_HR/"
- El script usado para generar las gráficas está en "scripts/generate_hidmarsh_rose_graphs.py"
- El script para ejecutar repetidamente y obtener datos del rendimiento está en "scripts/hindmarsh_rose.bash"
- La documentación está en "docs/html/index.html"