

## Apéndice A

# Herramientas numéricas

En el siguiente repositorio de GitHub se puede ver el código utilizado. También está presente un historial de los cambios, por si se quiere consultar el estado del código a fecha de presentación de este trabajo, en el supuesto de que haya cambios posteriores.

<https://github.com/trahald/EPES.git>

En el caso de no funcionar el enlace, contacte con el autor.

Principalmente se ha simulado usando el lenguaje de programación C debido a ser más eficiente numéricamente. El código ha sido escrito en su totalidad desde cero por el autor para la realización de este trabajo. No se ha reutilizado código de otros trabajos.

También se ha hecho uso de Python. Para la realización de las gráficas e imágenes se ha usado el módulo Matplotlib y ocasionalmente *gnuplot*. Para definir redes en Python hemos utilizado NetworkX y para la integrar la dinámica con un bajo numero de nodos, se ha usado SciPy y Numpy.

En el lenguaje C, para la integración de la dinámica se ha usado un algoritmo Runge-Kutta de orden 4. Para la percolación, se ha usado una modificación del algoritmo de Zeff-Newmann [13].

## Apéndice B

# Comentarios sobre la validación de las simulaciones

En este apéndice hacemos unos breves comentarios sobre la validación de las simulaciones.

Por un lado, tanto para la percolación como para la sincronización simulamos el caso base y lo comparamos con resultados de la literatura. Tanto para la percolación (Figura 2.4 del Capítulo 2) como para la sincronización (Figura 3.2 del Capítulo 3) los resultados coinciden con la literatura, luego razonamos que la implementación es correcta.

Para estar más seguros sobre, por ejemplo, la elección de paso temporal en la integración, integramos por un lado una función bien conocida, sin  $t$ , y comparamos con la solución analítica y hallamos el error. En la Figura B.1 mostramos los errores en la integración para distintos pasos temporales. Razonamos que un buen compromiso entre precisión y rapidez es  $h = 0.1$  o  $h = 0.01$ ;  $h = 0.1$  es el usado en la mayoría de las simulaciones del trabajo.  $h = 0.5$  para las simulaciones a alto  $\ell$  del Apéndice C.1 y en alguna simulación particular, cuando el acoplo es alto y la derivada es más sensible,  $h = 0.01$ .

También nos interesa conocer cuanto tiempo deberíamos esperar antes de empezar a promediar. Debemos asegurarnos de que el sistema está estacionario (es decir, se han borrado las condiciones iniciales) para que no tengamos sesgos en los promedios. En la Figura B.2 mostramos la evolución temporal para distintos acoplos en una red Erdős-Rényi con  $p = 0.15$ . Se observa que después de aproximadamente 4 “segundos”, o unidades de paso temporal, el sistema está termalizado. En las simulaciones siempre usaremos por los menos 10 unidades de tiempo antes de promediar para estar seguros.

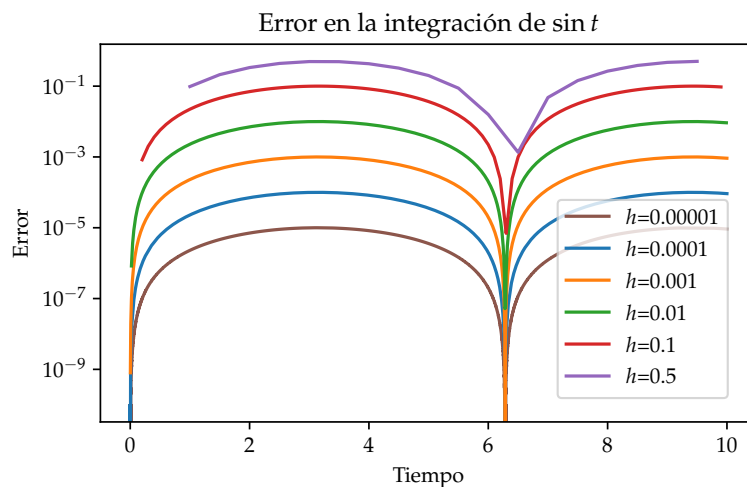


FIGURA B.1: Hemos integrado la función  $\sin(t)$ . En la gráfica mostramos el error en la integración al usar un algoritmo de Runge-Kutta de orden 4 para distintos pasos temporales.

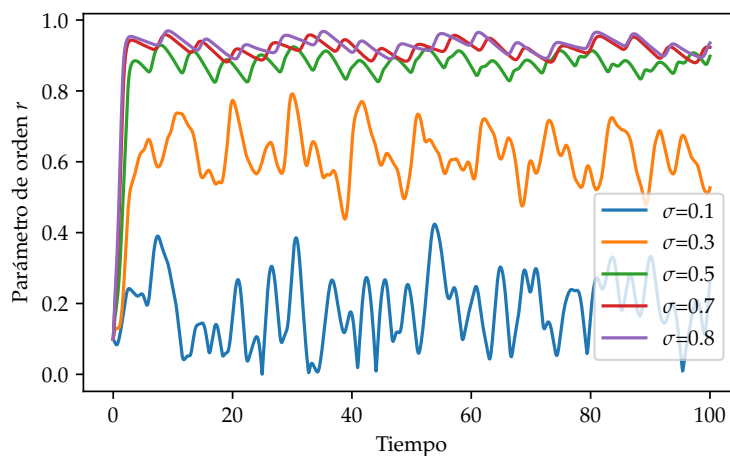


FIGURA B.2: Gráfica en la cual se muestra el parámetro de orden  $r$  en el tiempo para distintos valores de los acoplos. Se ha usado una red de  $N = 50$  y  $h = 0.001$ . La red sobre la cual se ha simulado la dinámica es una red aleatoria Erdős-Rényi con  $p = 0.15$ . Se observa que entorno para tiempos superiores a cuatro unidades, el sistema ha “termalizado”, entendiendo que se han perdido las condiciones iniciales. Para estar seguros, en las simulaciones siempre usamos un tiempo de termalización mayor de diez unidades temporales.

## Apéndice C

# Gráficas complementarias

### C.1. Comprobación del límite inferior para el acoplo

En este apéndice intentamos comprobar si existe algún límite inferior para los acoplos para los cuales hay sincronización explosiva, siguiendo el modelo del Capítulo 4. Para poder hacer la simulación en un tiempo razonable, usamos una red más pequeña con  $N = 500$  y un promediado de  $T = 10$  con paso temporal para la dinámica de osciladores de  $h = 0.5$ . Las frecuencias naturales se generan con una distribución plana en  $[-\frac{1}{2}, \frac{1}{2}]$ . Al ser acoplos muy débiles al módulo del vector velocidad  $|\frac{d\vec{\psi}}{dt}|$  de las frecuencias en el espacio de fases no será más grande que la máxima natural posible, luego podemos permitirnos un valor de  $h$  mayor. Usamos arbitrariamente el criterio  $|\frac{d\vec{\psi}}{dt}| \cdot h < 0.25$ . Aún disminuyendo la calidad de la dinámica integrada, se conserva la transición.

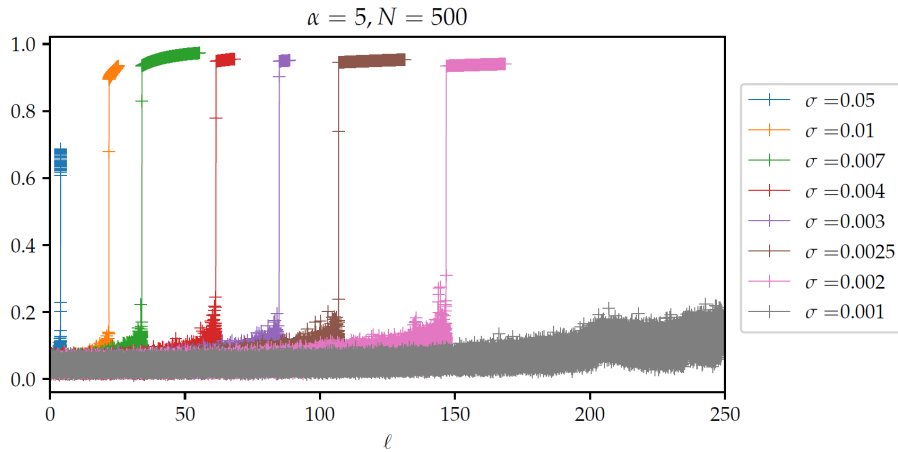


FIGURA C.1: (Versión digital en color.) Resultados de la simulación. Comentamos que no hemos hecho el proceso opuesto para observar la histéresis. Se observa que las transiciones desaparecen al bajar más allá de  $\sigma = 0.002$ .

Para una red de  $N = 500$ , tenemos el valor crítico para el acoplo en campo medio de  $\sigma_c = K_c/N = \frac{2}{\pi g(0) \cdot 500} = 0.00127$ . En el texto principal el mínimo valor de  $\sigma = K/N$  al que llegamos es  $\sigma = 0.01$ , aquí bajaremos un orden de magnitud más, hasta una milésima, pero para una red más pequeña. En la Figura C.1, la sincronización explosiva desaparece entre  $\sigma = 0.002$  y  $\sigma = 0.001$ , consistente con el valor calculado. Nótese que el valor de  $r$  es mucho más ruidoso. No es demostrable rigurosamente, pero en  $\ell \approx 205$  aparece un pequeño pico de una anchura de  $\Delta\ell = 10$ ; como la dinámica está promediada a una escala más corta de tiempo, podrían ser correlaciones en tiempo por no haber llegado a un estado estacionario. Podría ser interesante

realizar estas simulaciones para una red más grande, hallar numéricamente el punto crítico  $\sigma_c$ , y observar la transición que ocurre.

## C.2. Extensión del modelo para más valores de alfa

Consideramos que podría ser interesante ver la transición del modelo del Capítulo 4 a los distintos regímenes de  $\alpha$  de una manera más suave. En la Figura C.2 representamos varios valores intermedios de  $\alpha$ . La misma discusión del texto principal es válida aquí, pero se facilita la visualización con más valores intermedios.

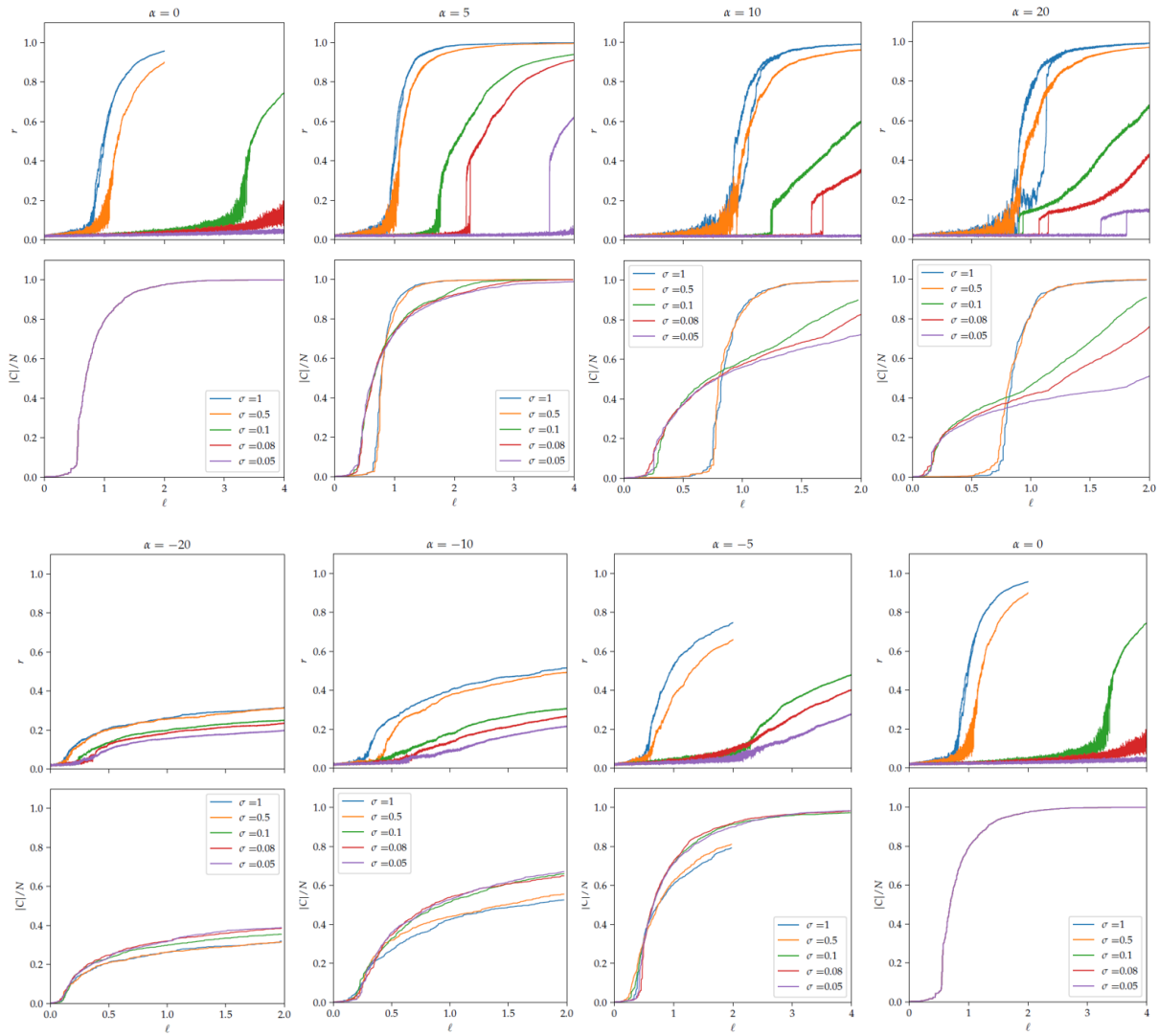


FIGURA C.2: El modelo del Capítulo 4 para más valores intermedios de  $\alpha$  respecto de la gráfica del texto principal, usando los mismos parámetros para la simulación.