

Bright solitons

1. Programas

Los archivos correspondientes al caso de *bright solitons* se han modificado de manera que ahora tenemos tres partes diferenciadas: el input, el cálculo y generación de archivos y finalmente, la salida de gráficas y otros. Tenemos pues los siguientes archivos:

- *input.txt* : se corresponde con el archivo de entrada de datos. Está ajustado de forma similar a lo que saldría de la introducción de parámetros de una interfaz, aunque con un título encima que nos muestra qué parámetros se corresponden con cada columna.
- *gpe_bright_solitons.py* : contiene el programa original pero modificado de tal forma que no se muestra ningún resultado o dato por pantalla. Este llama a su vez a distintos módulos para proceder a los cálculos y generación de archivos con datos.
- *bs_plots.py* : este programa recoge los datos guardados en archivos creados por el programa anterior y los usa para crear gráficos que muestren distintas situaciones. También importa la información de uno de los módulos del programa principal para poder dar los gráficos adecuados a cada situación.

1.1. Input

En este caso se han dejado como parámetros a introducir los mismos que se pedían por pantalla en el programa original. Aún así, se ha puesto la velocidad como un float directamente para poder probar distintos casos, y por lo visto no genera ningún problema en cuanto a conservación de energía. Está sujeto a valoración si se deja como float o como int (introducción por el usuario).

A su vez, los valores (excepto la velocidad como se ha comentado) presentan las mismas opciones que en el programa original. No obstante, en el input creado se han recogido todos los posibles casos en una entrada de datos. No hay problema por eso, puesto que el uso de las variables *wb* y *hb* sólo se usarán en caso que el potencial sea el adecuado.

1.2. Cálculo y generación de datos

En el código no se muestran posibles errores de introducción de parámetros pues se supone que la interfaz ya limitará esos problemas, no cómo si la introducción fuera manual por el usuario.

Por lo que respecta al archivo *gpe_bright_solitons.py* este ya no incluye tampoco la creación de la animación, y se impone la creación de archivos y la no representación instantánea de

figuras. Así, este programa llama a distitos módulos que generarán en conjunto los archivos. El primer módulo es el de parámetros, que recoge los datos introducidos en *input.txt* y asigna a las variables que dependen de estos un valor determinado; también fija ciertos parámetros involucrados en los cálculos (Zmx , $Npoint$, etc.).

El archivo de *gpe_bs_utilities.py* apenas ha sido modificado, sólo anulando los comentarios que se imprimían por pantalla. Y por lo que respecta a *gpe_bs_plots.py* no se ha modificado nada porque ya no se usará, puesto que los gráficos se harán de forma independiente a los cálculos.

El archivo correspondiente a evolución sí que se ha modificado, pero es más respecto a eliminar todo lo que se correspondía con gráficas y datos por pantalla. Se ha dejado la escritura de resultados en archivos externos, así como se han creado nuevos archivos de outputs correspondientes al valor medio de la función de onda del solitón (en este caso se ha considerado por el punto más alto de la función de onda) con respecto al tiempo, para poder luego hacer una nueva gráfica que facilite la visualización; en este caso sólo se han escogido unos puntos determinados, para hacer así algo más limpio y claro a la vista del usuario.

1.3. Plots

Finalmente, en *bs_plots.py* se hacen las representaciones. Entrando los archivos generados antes, se elaboran matrices para guardar los datos que nos interesan, y posteriormente se procede a su representación.