

UNIVERSIDAD NACIONAL DE EDUCACIÓN A DISTANCIA
FACULTAD DE CIENCIAS - FÍSICA FUNDAMENTAL
MÁSTER EN FÍSICA DE SISTEMAS COMPLEJOS

TRABAJO FIN DE MÁSTER

Investigación del problema de Fermi-Pasta-Ulam

Memoria realizada por: Víctor Borrero Mayora
Tutor: Prof. Dr. Ignacio Zúñiga López



UNIVERSIDAD NACIONAL DE EDUCACIÓN A DISTANCIA
FACULTAD DE CIENCIAS - FÍSICA FUNDAMENTAL
MÁSTER EN FÍSICA DE SISTEMAS COMPLEJOS

TRABAJO FIN DE MÁSTER

Investigación del problema de Fermi-Pasta-Ulam

Memoria realizada por: Víctor Borrero Mayora
Tutor: Prof. Dr. Ignacio Zúñiga López



Índice general

I	Comunes	8
	Agradecimientos	10
	Resumen	11
	Apuntes	12
II	Memoria	12
1.	Introducción	13
	1.1. El problema original	13
	1.2. Los estudios posteriores	17
	1.3. Resolución Numérica del problema FPU en MATLAB	19
	1.4. Los descubridores	21
2.	Formulación del Problema Lineal	24
	2.1. Definición del Problema Lineal	24
	2.2. Resultados Gráficos	30
3.	Formulación del Problema de Fermi-Pasta-Ulam	32
	3.1. Hamiltoniano de Fermi-Pasta-Ulam	32
	3.2. Equipartición de la Energía	36
	3.3. Creación del algoritmo	37
4.	Desarrollos del Problema FPU	42
	4.1. Los desarrollos iniciales a partir del problema FPU	42
	4.2. La teoría de los solitones	44
	4.3. La red discreta y la teoría de la perturbación	47
	4.4. La teoría de Kolmogorov-Arnold-Moser	48
	4.5. La red de Toda	49

5. Resultados de la Simulación del Problema FPU	51
5.1. El programa en MATLAB	52
5.2. Ejecución del programa	55
5.3. Presentación de los resultados	57
III Conclusión	68
1. Conclusión	69
IV Bibliografía	73
Bibliografía	79
V Apéndices	80
A. Programas	81
B. Archivos	87
C. Etapas de Desarrollo del Proyecto	90
C.1. Etapas	90
C.2. Diagrama de Gantt	92
D. Presupuesto y Material	93
D.1. Material	93
D.2. Presupuesto	94

Índice de figuras

II	Memoria	13
1.1.	El algoritmo de Mary Tsingou para el primer experimento numérico en el MANIAC I, con fecha de 20 de mayo de 1955	14
2.1.	Cadena de osciladores	24
2.2.	La energía se mantiene constante en el problema lineal	31
2.3.	Las diferencias oscilan siguiendo la condición inicial	31
4.1.	Cadena de osciladores	43
5.1.	En el problema FPU la energía se reparte entre los modos y después vuelve al modo inicial	57
5.2.	En las diferencias espaciales se aprecia la aparición de solitones en el problema FPU	58
5.3.	Extracción de los solitones en el problema FPU	59
5.4.	Vídeo de los solitones en el problema FPU	59
5.5.	Aparición de la super recurrencia en el problema FPU	60
5.6.	Energía de los modos dependiendo de la amplitud	61
5.7.	Los solitones en la red de Toda	62
5.8.	Tiempos de recurrencia para diferentes valores de N	64
5.9.	Estudio logarítmico del período de recurrencia	65
5.10.	Estructuras con la condición inicial de Zabusky-Deem	66

III	Conclusión	69
IV	Bibliografía	74
	Apéndices	81
	B.1. Lista de archivos adjuntos	89
	C.1. Diagrama de Gantt del Proyecto Fin de Máster	92
	D.1. Ordenador portátil <i>HP EliteBook 820</i>	94

Índice de tablas

5.1. Tiempo de recurrencia analítico con $a = 1$ y $\alpha = 0,25$	64
5.2. Tiempo de recurrencia numérico con $a = 1$ y $\alpha = 0,25$	65

Parte I
Comunes

Agradecimientos

Este Trabajo de Fin de Máster, como siempre, se lo dedico a mi familia, por su apoyo y amor incondicionales, y en especial a mi mujer Carol. También me gustaría agradecer su ayuda al profesor D. Ignacio Zúñiga de la *Facultad de Ciencias de la UNED* y a mis antiguos profesores por despertar en mí la curiosidad y el amor por la Ciencia.

Resumen

En este trabajo se ha llevado a cabo el estudio del problema de Fermi-Pasta-Ulam recopilando información sobre su origen y formulación así como simulaciones numéricas para su análisis.

Este problema que, a priori, parecía sencillo, de ahí que fuese seleccionado para realizar la primera simulación numérica de la historia de la Física, sigue al cabo de los años aportando nuevas ideas y despertando nuevas líneas de investigación en múltiples ámbitos de la Física tanto teórica como experimental y es por ello que su estudio sigue a día de hoy siendo un tema apasionante para los amantes de la Física.

Para el análisis numérico se ha partido de un código en Matlab que se ha modificado para permitir la simulación del problema obteniendo los mismos resultados que en su día obtuvieron Fermi, Pasta, Ulam (y Tsingou) en el laboratorio de Los Álamos, y también incluyendo desarrollos posteriores. De esta manera se ha creado una herramienta que sirve para analizar el problema y para ilustrar varias de las teorías surgidas a partir del problema FPU.

Además se han recopilado los artículos originales sobre el problema FPU para que puedan usarse como referencias para futuros análisis, y también se han incluido artículos sobre los nuevos desarrollos. De esta manera se ha creado una base bibliográfica que puede consultarse para el estudio del problema FPU, tanto en sus orígenes como en la época actual.

Descriptores

FPU, Fermi-Pasta-Ulam, FPUT, Fermi-Pasta-Ulam-Tsingou

Apuntes

Este trabajo se ha estructurado siguiendo las secciones de un artículo científico, con introducción, formulación, desarrollos, resultados, conclusiones, bibliografía y apéndices. En los apéndices se ha incluido una parte sobre la Planificación y el Presupuesto necesario para completar el trabajo para así explicitar la metodología que se ha utilizado.

Parte II
Memoria

CAPÍTULO 1

Introducción

El problema de Fermi-Pasta-Ulam, conocido también como FPU, tiene unos interesantísimos orígenes. Como su nombre indica los tres científicos que descubrieron este fenómeno fueron Enrico Fermi, John Pasta y Stanislaw Ulam tras la publicación de un informe en el laboratorio de Los Álamos en 1955 ([1]). Este informe fue el precursor de dos novedosas ramas científicas: la Física no lineal y el cálculo computacional para la simulación y resolución de problemas.

1.1 El problema original

Tras la Segunda Guerra Mundial Fermi solía visitar con frecuencia Los Álamos ([2]), el laboratorio fundado durante la guerra para coordinar el desarrollo de las armas nucleares del Proyecto Manhattan, y, al principio de la década de los cincuenta, Fermi empezó a considerar el enorme potencial de las computadoras no sólo para la realización de cálculos sino también para la simulación de problemas físicos. Ése fue el origen de los experimentos numéricos.

En 1952 se había instalado en Los Álamos una de las primeras computadoras, conocida como MANIAC I (Mathematic Analyzer, Numerical Integrator And Computer). En aquel entonces prácticamente

Fermi tenía el convencimiento que las nuevas teorías fundamentales de la Física contendrían operadores y ecuaciones no lineales y que por lo tanto podía ser extremadamente útil adquirir los conocimientos matemáticos necesarios para entender el comportamiento de sistemas no lineales ([2]). Por ello el objetivo era realizar simulaciones numéricas de problemas sin soluciones analíticas explícitas para así poder estudiar si esas soluciones permitían entender mejor el comportamiento del sistema, en especial en lo tocante a la conducta asintótica o a largo plazo de problemas no lineales.

Para ello, Fermi, Pasta y Ulam decidieron seleccionar el modelo físico más sencillo que cumpliera estos requisitos y simular su comportamiento a largo plazo, que resultó ser el estudio de la evolución de un cristal hacia el equilibrio térmico modelizado por medio de una cadena de partículas u osciladores de masa unitaria unidos por un potencial de interacción cuadrático (esto se correspondería con un sencillo problema que puede resolverse analíticamente) pero también con una interacción débil no lineal (lo que impide la resolución analítica del problema). De esta manera se pasa de un modelo continuo (el cristal con los extremos fijos) a un modelo discreto (las masas unitarias o puntos) lo cual permite transformar la ecuación diferencial del sistema por un conjunto de ecuaciones diferenciales que a su vez se convierte en ecuaciones de diferencias para la simulación numérica.

Tomando esto como punto de partida, un problema unidimensional, el objetivo era continuar después con problemas más complejos como, por ejemplo, el mismo problema pero bidimensional, para después pasar a modelos de sistemas con mezcla y con turbulencias. De hecho se menciona explícitamente ([1]) que ese informe iba a ser el primero de una serie de informes sobre el comportamiento de sistemas no lineales en los que la característica no lineal se ha introducido como una perturbación de un problema original lineal.

Las primeras conversaciones tuvieron lugar durante el verano del año 1952 y las primeras simulaciones un año después en el verano de

1953. El informe ya estaba casi completado en 1954 pero en noviembre de ese año falleció Fermi y los últimos cálculos los terminaron sin él publicando el informe secreto en mayo de 1955; de ahí que la serie de informes no llegase a completarse.

En el modelo lineal la solución del problema es una vibración periódica de la cuerda (o las masas). Si la posición inicial es una única curva sinusoidal entonces la cuerda oscilará siempre en ese modo. El objetivo de la simulación era ver como se comportaría el sistema con el componente no lineal. El resultado esperado por Fermi era que la cuerda adquiriría formas cada vez más complejas y con el paso del tiempo todos los modos adquirirían cada vez más importancia. Si, por ejemplo, el punto de partida era un único modo entonces la cuerda derivaría poco a poco hacia los demás modos hasta que se obtuviese la equipartición de la energía que predice la Mecánica Estadística.

Al principio de las simulaciones numéricas excitando primero un modo y después otro los resultados numéricos obtenidos parecían confirmar esta hipótesis. En la simulación original en la computadora MANIAC I la simulación se hacía con 64 puntos. Los cálculos tardaban mucho en completarse y los resultados que se observaban antes de detener el experimento parecían confirmar la equipartición, pero un día que se dejó la computadora funcionando por error y al comprobarse los resultados obtenidos en esa simulación más larga de lo previsto se observó algo inesperado: aparecía un comportamiento casi periódico en el que prácticamente toda la energía original regresaba al modo inicial y casi se recuperaba el estado inicial. Tras 157 períodos del modo $k = 1$ el 97 % de la energía regresaba al modo original ([4]).

Por lo tanto, contrariamente a lo esperado por Fermi, Pasta y Ulam el reparto de la energía del modo original entre los demás modos no ocurre. Este sorprendente resultado es lo que se conoce como la paradoja de FPU: la no linealidad no es condición suficiente para garantizar la equipartición de la energía.

1.2 Los estudios posteriores

Tras la publicación de sus resultados en 1955 el informe empezó a circular pero de forma restringida a causa del fallecimiento de Fermi y al tratarse de un informe del laboratorio de Los Álamos. En 1965 al publicarse la obra completa de Fermi el problema FPU resurgió con más fuerza y empezaron a realizarse más estudios sobre el tema, aunque ya en 1961 Tuck y Menzel utilizaron computadoras más modernas y más potentes para investigar el comportamiento del problema FPU al cabo de más tiempo. De esta manera se dieron cuenta de que tras la primera recurrencia que conllevaba una pérdida de energía de alrededor del 1 % del modo principal, si se continúa el cálculo, el modo principal sigue perdiendo energía hasta que al cabo de 8 ciclos va recuperando energía ciclo a ciclo y finalmente a los 16 ciclos se obtiene un pico cuya energía en el modo principal es prácticamente igual a la original. Esto pasó a denominarse como superciclo ([5]).

También en 1961, Zabusky se interesó por la resolución del problema de forma analítica ([6]) consiguiendo resultados para el problema continuo que concordaban con los resultados numéricos hasta que se alcanzaba una singularidad. Zabusky continuó estos trabajos con Kruskal, trabajando en la resolución en el espacio real en vez del espacio de Fourier ([7]). En ese trabajo, para explicar el comportamiento periódico, introdujeron el concepto de solitón, es decir de ondas no lineales, estables, localizadas o solitarias que se comportan como partículas y que aparecen en la ecuación de Korteweg-de Vries (KdV), habiendo demostrado que esta última constituye una aproximación continua de primer orden del problema FPU. Ese descubrimiento fue el punto de partida de lo que ahora se conoce como la Física de solitones ([3]).

Además de estos trabajos sobre solitones también se trabajó sobre la dinámica de los modos de Fourier. En especial con el descubrimiento del teorema de Kolmogorov-Arnold-Noser que demostró que en general las órbitas de sistemas hamiltonianos integrables que han sido ligeramente perturbados son casi-periódicas. Pero si la perturbación es demasiado fuerte entonces la recurrencia desaparece y la equipartición de la energía

se establece rápidamente como queda reflejado en [3].

En paralelo a todo este trabajo en Japón se estaba desarrollando una investigación similar pero sin tener acceso a los documentos publicados en Occidente. En especial Nobuhiko Saitô y su estudiante Hajime Hirooka trabajaban en la tesis del segundo titulada “Acercamiento al equilibrio térmico en una red no lineal realizando sus propias simulaciones numéricas entre 1964 y 1965 sin conocer los resultados obtenidos por Fermi, Pasta y Ulam ([4]). Dada la paradoja de sus resultados no se decidieron a publicarlos hasta 1967 ([8] y [9]), una vez habían leído los trabajos de Fermi, Pasta, Ulam de 1955 ([1]) y de Ford de 1961 ([10]). Al mismo tiempo Morikazu Toda tuvo acceso en Japón a los trabajos sobre la paradoja FPU y empezó a interesarse por el problema publicando también en 1967 un trabajo sobre una red con interacciones exponenciales conocida en la actualidad como red de Toda ([11]). Además en 1968 Zabusky viajó a Kioto para una conferencia y presentó su trabajo sobre la ecuación KdV y justo en aquella época Toda publicó su fórmula analítica para la estimación del período de recurrencia, T_R , cuyo valor era muy próximo a la fórmula empírica obtenida por Zabusky ([4]).

Así pues tanto en EE.UU. como en Japón el problema FPU pronto se convertiría en un tema con amplios desarrollos. La paradoja de Fermi-Pasta-Ulam es pues el punto de partida de la Física computacional, de la teoría de los solitones y de la teoría del caos, y también uno de los primeros ejemplos de estudio de la teoría de KAM, y fue por lo tanto un trabajo clave para el desarrollo de la Física moderna.

1.3 Resolución Numérica del problema FPU en Matlab

En este trabajo de fin de Máster reproduciremos los resultados originales y también trabajaremos sobre preguntas y respuestas que surgieron con posterioridad. Para ello utilizaremos como punto de partida las preguntas formuladas por T. Dauxois, M. Peyrard y S. Ruffo en [4]. En ese artículo también aparece un código en MATLAB para obtener los resultados del problema FPU. Ese código ha sido modificado y ampliado considerablemente para facilitar la ejecución, la comprensión y para la obtención de resultados y gráficas.

Como puede verse en el apéndice, que contiene el código modificado, con las actuales herramientas informáticas lo que en los años 50 requería un complejo algoritmo con ejecuciones que duraban horas o días, ahora puede conseguirse con un ordenador personal y con un programa relativamente corto y manejable.

Si nos referimos a [4], las preguntas planteadas son las siguientes:

- En primer lugar, reproducir los resultados originales del experimento de Fermi, Pasta y Ulam. Hay que ser cuidadoso con la amplitud a puesto que el término cúbico se vuelve inestable con niveles altos de energía.
- Analizando el campo de diferencias de desplazamiento de dos masas consecutivas, $(u_{i+1} - u_i)$, pueden apreciarse estructuras que se propagan, que se corresponden con los solitones del análisis de Zabusky-Kruskal que puede leerse en [7].
- Para amplitudes entre $a = 1$ y $a = 10$, el modelo FPU con 32 masas, $N = 32$, se relaja hasta el equilibrio ([12]). Puede verificarse que no se dan las recurrencias del modelo FPU y que la energía se reparte entre los modos normales.
- Puesto que la onda sinusoidal inicial forma solitones ([7]) es lógico plantear el problema usando como condición inicial una red de solitones. La solución exacta de solitones tipo *kink* para la red de Toda

([13]) es

$$u_i(t) = \pm \frac{1}{2\alpha} \log \left(\frac{1 + \exp[2k(i - 1 - i_0 \pm t \sin(hk))]}{1 + \exp[2k(i - i_0 \pm t \sin(hk))]} \right), \quad (1.1)$$

con k la inversa del ancho. Esta solución es solamente aproximada para la red FPU, y a causa de las condiciones fijas de frontera, debe ponerse una solución *kink* (+) y una solución *antikink* (-), centradas en lados opuestos de la red. Para poder simular este problema se recomienda el uso de una red fina, por ejemplo $N = 128$. La evolución temporal del campo diferencial $u_{i+1} - u_i$ muestra la propagación de estructuras regulares, su interacción mutua y el retorno al alcanzar las fronteras.

- Seguidamente, confirmar que efectivamente el período de recurrencia, T_R , se comporta como indica la fórmula de Toda que aparece en [13], con el período del primer modo $T_1 = 2N$,

$$T_R = \frac{3}{\pi^{3/2}\sqrt{2}} \frac{N^{3/2}}{\sqrt{a\alpha}} T_1 \simeq 0,76 \frac{N^{5/2}}{\sqrt{a\alpha}}, \quad (1.2)$$

es decir que el período de recurrencia evoluciona como una potencia del número de osciladores en $N^{5/2}$. Esta fórmula ha sido modificada con respecto a [3] para solventar un error numérico.

- Finalmente, usar la condición inicial de Zabursky-Deem que aparece en [14], $u_i = a \sin[i\pi N/(N + 1)]$, y que se corresponde con el modo de frecuencia más alta. En este caso se observan oscilaciones de los modos de energía similares a las de longitudes de onda grandes. Se intentará caracterizar estas recurrencias.

Con esta serie de preguntas por responder el trabajo utilizará la bibliografía existente y las simulaciones numéricas para conseguir los resultados.

1.4 Los descubridores

En esta sección se incluyen pequeñas biografías de los investigadores principales que participaron en el descubrimiento de la paradoja FPU, es decir Fermi, Pasta, Ulam y también Tsingou.

1.4.1 Enrico Fermi

Enrico Fermi (Roma, 29 de septiembre de 1901-Chicago, 28 de noviembre de 1954) fue uno de los físicos más relevantes del siglo XX. Tuvo contribuciones relevantes en múltiples áreas como la Física Cuántica, Física Nuclear, Física de Partículas y Mecánica Estadística. Se doctoró por la Scuola Normale Superiore de Pisa en 1922 convirtiéndose en 1927 en profesor de la Universidad de Roma “La Sapienza”. Recibió el premio Nobel de Física en 1938, por sus contribuciones sobre nuevos elementos radioactivos, circunstancia que aprovechó para dejar la Italia de Mussolini y emigrar a EE.UU. convirtiéndose en profesor de la Universidad de Columbia en Nueva York y de la Universidad de Chicago, trabajando principalmente en física nuclear. En este período llevó a cabo la construcción de la primera pila nuclear y consiguió la primera reacción en cadena controlada en Chicago en 1942. Durante el resto de la Segunda Guerra Mundial trabajó en el proyecto Manhattan en el laboratorio de Los Álamos desarrollando la primera bomba atómica.

Si bien es cierto que su trabajo en el origen de la Física no lineal y el cálculo computacional no ha tenido tanta relevancia como sus otras contribuciones, con el paso del tiempo el problema FPU se ha convertido también en algo fundamental que agranda más si cabe su leyenda como una de los físicos principales del siglo XX, junto a Einstein, Born, Dirac, Schrödinger, Pauli y Feynman entre otros. Su importancia en la Física del siglo XX queda patente por la gran cantidad de elementos dedicados a Fermi, como los fermiones, el Fermilab, el elemento químico Fermio, la estadística de Fermi-Dirac o la paradoja de Fermi-Pasta-Ulam.

1.4.2 John R. Pasta

John Robert Pasta (Nueva York, 22 de octubre de 1918-Chicago, 5 de junio de 1984) fue un físico computacional americano. Tras comenzar sus estudios en el City College de Nueva York tuvo que abandonarlos a causa de la depresión de los años 30 y se puso a trabajar primero como agente inmobiliario y después como agente de policía. En 1942 fue reclutado por el ejército de EE.UU. y trabajó como oficial de señales. Tras la Segunda Guerra Mundial terminó sus estudios universitarios y completó su doctorado en la Universidad de Nueva York. En 1951 se incorporó al laboratorio de Los Álamos.

En Los Álamos trabajó en la computadora MANIAC I y formó parte del equipo que descubrió la paradoja FPU. Tras este proyecto siguió trabajando con ordenadores en la Comisión para la Energía Atómica de EE.UU. y después como profesor en la Universidad de Illinois.

1.4.3 Stanislaw M. Ulam

Stanislaw Marcin Ulam (Lviv, 13 de abril de 1909-Santa Fe, 13 de mayo de 1984) fue un matemático polaco-americano conocido por el desarrollo de las armas termonucleares, el método de Monte Carlo y la paradoja FPU. Nacido en el seno de una rica familia judeo-polaca, estudió matemáticas y se doctoró en el Instituto Politécnico de Lwów, trabajando con el gran matemático polaco Stefan Banach. En 1935 John von Neumann, a quien había conocido en Varsovia, le invitó a trabajar unos meses en Princeton. Posteriormente entre 1936 y 1939 pasó el curso académico en Harvard trabajando en la teoría ergódica para finalmente establecerse en los EE.UU. como profesor en la Universidad de Wisconsin-Madison, convirtiéndose en estadounidense en 1941.

En 1943 se incorpora al proyecto Manhattan, y tras el fin de la guerra se incorpora como profesor a la Universidad de Southern California pero un ataque de encefalitis en 1946 que necesitó una cirugía cerebral hace que deje ese puesto y decida volver al laboratorio de Los Álamos para trabajar en el desarrollo de las armas termonucleares, co-

laborando así con varias de las personas que le habían visitado durante su convalecencia; en 1951 ideó el conocido como diseño de Teller-Ulam que constituye la base de las armas termonucleares. Empezó a trabajar con computadoras en Los Álamos y de hecho el método de Monte Carlo fue desarrollado por él y por Nicholas Metropolis (su nombre proviene del hecho que el tío de Stanislaw, Michael Ulam, pasaba largas estancias en Monte Carlo) y poco después participó en el descubrimiento de la paradoja FPU que fue el germen de la física no lineal. Siguió trabajando en Los Álamos hasta 1965 y después se convirtió en profesor de matemáticas en la Universidad de Colorado.

1.4.4 Mary Tsingou

Mary Tsingou (Milwaukee, 14 de octubre de 1928) es una física y matemática americana. Se graduó por la Universidad de Wisconsin en 1951 completando sus estudios en la Universidad de Michigan. En aquella época era poco habitual que las mujeres trabajaran como matemáticas pero a causa de la falta de hombres por la guerra de Corea y los fallecimientos en la Segunda Guerra Mundial consiguió un puesto en 1952 en el laboratorio de Los Álamos para realizar cálculos.

Fue entonces cuando comenzó su trabajo con la MANIAC I y participó en la creación de los algoritmos y la codificación de la paradoja de FPU. Posteriormente trabajó en múltiples problemas computacionales, siendo una de las primeras personas en aprender Fortran, lenguaje inventado por IBM en 1955. En los 70 volvió a estudiar con Tuck el problema FPU con simulaciones más largas, también trabajó en la solución numérica de las ecuaciones de Schrödinger y colaboró con John von Neumann en el estudio de la mezcla de dos fluidos con densidades diferentes. En los 80 trabajó en el proyecto *Star Wars*, sobre armas en satélites. Se jubiló en 1991 y aún reside cerca de Los Álamos.

Recientemente han surgido voces que propugnan que el problema FPU debería rebautizarse como problema FPUT, por Fermi, Pasta, Ulam y Tsingou, para que su participación en el estudio original quede claramente reflejada, pero todavía no se ha popularizado este nombre.

CAPÍTULO 2

Formulación del Problema Lineal

En esta sección introduciremos las nociones principales del problema lineal a partir del cual surgió el problema de Fermi-Pasta-Ulam.

2.1 Definición del Problema Lineal

Supongamos que tenemos una cadena de N partículas de masa m , en línea, estando cada partícula atada a las dos partículas que la preceden y suceden por una fuerza de Hooke de constante κ . Supongamos que las partículas sólo pueden desplazarse en una dirección y al desplazamiento de la partícula i a partir de la posición de equilibrio lo denominaremos u_i ; además supondremos que los extremos están fijos lo cual modelizaremos estableciendo que los desplazamientos u_0 y u_{N+1} son nulos para las partículas 0 y $N + 1$.

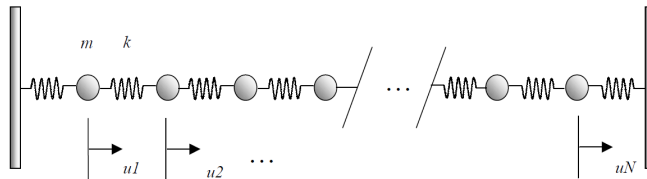


Figura 2.1: Cadena de osciladores

La ecuación de movimiento de la partícula i se escribiría como

$$\begin{aligned} m \frac{d^2}{dt^2} u_i(t) &= \kappa (u_{i+1}(t) - u_i(t)) - \kappa (u_i(t) - u_{i-1}(t)) \\ m \frac{d^2}{dt^2} u_i(t) &= \kappa (u_{i+1}(t) - 2u_i(t) + u_{i-1}(t)) \end{aligned} \quad (2.1)$$

y en términos de hamiltoniano la ecuación se escribiría como

$$\mathcal{H} = \sum_{i=1}^N \frac{p_i^2}{2m} + \sum_{i=0}^N \frac{\kappa}{2} (u_{i+1} - u_i)^2, \quad (2.2)$$

con

$$\begin{aligned} \dot{u}_i &= \frac{\partial \mathcal{H}}{\partial p_i} = \frac{p_i}{m} \\ \dot{p}_i &= -\frac{\partial \mathcal{H}}{\partial u_i} = \kappa (u_{i+1} - 2u_i + u_{i-1}) \end{aligned} \quad (2.3)$$

Estas ecuaciones pueden reescribirse en forma matricial como

$$\frac{d^2}{dt^2} \mathbf{u}(t) = -\omega_0^2 \mathbf{K} \mathbf{u}(t) \quad (2.4)$$

con $\omega_0^2 = \kappa/m$ y \mathbf{K} una matriz tridiagonal

$$\mathbf{K} = \begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{bmatrix}. \quad (2.5)$$

Si ahora hacemos un cambio de variable (o una transformada de Fourier) y escribimos

$$\mathbf{u}(t) = \mathbf{v} \cos(\omega t + \phi), \quad (2.6)$$

entonces obtenemos un problema de autovalores que se escribe como

$$\mathbf{K} \mathbf{v} = \lambda \mathbf{v} \quad (2.7)$$

con $\lambda = \omega^2/\omega_0^2$. Con la matriz tridiagonal de este sistema es posible calcular los autovalores y autovectores de forma analítica. Los autovectores serán las columnas de la matriz de transformación \mathbf{V} cuyos coeficientes se escribirán como

$$V_{ik} = \sqrt{\frac{2}{N+1}} \sin\left(\frac{ik\pi}{N+1}\right), \quad (2.8)$$

y los autovalores serán

$$\omega_k = 2\sqrt{\frac{\kappa}{m}} \sin\left(\frac{k\pi}{2(N+1)}\right). \quad (2.9)$$

Esta matriz de transformación es ortogonal y diagonaliza la matriz \mathbf{K} . Así pues se pueden introducir coordenadas normales

$$\mathbf{A} = \sqrt{m}\mathbf{V}^T \mathbf{u}, \quad (2.10)$$

con la relación inversa

$$\mathbf{u} = \frac{1}{\sqrt{m}} \mathbf{V} \mathbf{A}, \quad (2.11)$$

y el momento se escribirá

$$\mathbf{P} = \frac{1}{\sqrt{m}} \mathbf{V}^T \mathbf{p}, \quad (2.12)$$

y por lo tanto

$$\mathbf{p} = \sqrt{m}\mathbf{V}\mathbf{P}. \quad (2.13)$$

Con esta nueva notación podemos reescribir el hamiltoniano en forma matricial como

$$\mathcal{H} = \frac{1}{2m} \mathbf{p}^T \mathbf{p} + \frac{\kappa}{2} \mathbf{u}^T \mathbf{K} \mathbf{u} \quad (2.14)$$

que puede reescribirse como

$$\mathcal{H} = \frac{1}{2} \sum_{k=1}^N [P_k^2 + \omega_k^2 A_k^2], \quad (2.15)$$

o teniendo en cuenta la relación entre p_i y u_i perfilada anteriormente

$$\mathcal{H} = \frac{1}{2} \sum_{k=1}^N \left[\frac{1}{m} \dot{A}_k^2 + \omega_k^2 A_k^2 \right], \quad (2.16)$$

y en términos de energía se escribiría

$$\mathcal{H} = \sum_{k=1}^N E_k, \quad (2.17)$$

con

$$E_k = \frac{1}{2} \left[\frac{1}{m} \dot{A}_k^2 + \omega_k^2 A_k^2 \right]; k = 1, 2, \dots, N. \quad (2.18)$$

Las ecuaciones de movimiento se escribirían

$$\frac{d^2}{dt^2} A_k(t) + \omega_k^2 A_k(t) = 0; k = 1, 2, \dots, N \quad (2.19)$$

cuya solución es

$$A_k(t) = A_k(0) \cos(\omega_k t) + \dot{A}_k(0) \frac{1}{\omega_k} \sin(\omega_k t), \quad (2.20)$$

y los modos normales se escribirán

$$A_k = \sqrt{m} \sqrt{\frac{2}{N+1}} \sum_{i=1}^N u_i \sin\left(\frac{ik\pi}{N+1}\right) \quad (2.21)$$

$$u_i = \frac{1}{\sqrt{m}} \sqrt{\frac{2}{N+1}} \sum_{k=1}^N A_k \sin\left(\frac{ik\pi}{N+1}\right)$$

y

$$P_k = \frac{1}{\sqrt{m}} \sqrt{\frac{2}{N+1}} \sum_{i=1}^N p_i \sin\left(\frac{ik\pi}{N+1}\right) \quad (2.22)$$

$$p_i = \sqrt{m} \sqrt{\frac{2}{N+1}} \sum_{k=1}^N P_k \sin\left(\frac{ik\pi}{N+1}\right)$$

De esta manera vemos que se consiguen soluciones analíticas y que se corresponden a la suma de los modos normales de oscilación.

Ahora podemos reescribir la solución en términos de la Transformada de Fourier Discreta, \mathcal{DF} . Para ello vamos a transformar la ecuación de los factores A_k . Así pues tenemos

$$\begin{aligned} A_k &= \sqrt{m} \sqrt{\frac{2}{N+1}} \sum_{i=1}^N u_i \sin\left(\frac{ik\pi}{N+1}\right) \\ &= \sqrt{m} \sqrt{\frac{2}{N+1}} \sum_{i=0}^N u_i \sin\left(\frac{ik2\pi}{2(N+1)}\right), \end{aligned} \quad (2.23)$$

con lo cual introducimos el valor 2π que aparece en la Transformada de Fourier y usamos el hecho de que $u_0 = 0$ para añadir el índice 0. Ahora podemos introducir la parte imaginaria para transformar el seno. Tenemos

$$A_k = \sqrt{m} \sqrt{\frac{2}{N+1}} \Im \left[\sum_{i=0}^N u_i e^{\frac{ik2\pi j}{2(N+1)}} \right]. \quad (2.24)$$

Ahora vamos a utilizar una propiedad de la exponencial. La exponencial es simétrica conjugada con respecto al punto central en la transformada de Fourier. Eso es fácil de ver puesto que $\Im[e^{\frac{ik2\pi j}{2(N+1)}}] = \Im[-e^{\frac{(2(N+1)-i)k2\pi j}{2(N+1)}}]$ puesto que $e^{ik2\pi j} = 1$ si el producto ik es un número entero. Esto nos permite escribir que

$$\Im \left[u_i e^{\frac{ik2\pi j}{2(N+1)}} \right] = \frac{1}{2} \Im \left[u_i e^{\frac{ik2\pi j}{2(N+1)}} - e^{\frac{(2(N+1)-i)k2\pi j}{2(N+1)}} \right]. \quad (2.25)$$

Finalmente en este punto podemos reescribir la ecuación anterior como

$$A_k = \sqrt{m} \sqrt{\frac{1}{2(N+1)}} \Im \left[\sum_{i=0}^N \left(u_i e^{\frac{ik2\pi j}{2(N+1)}} - u_i e^{\frac{(2(N+1)-i)k2\pi j}{2(N+1)}} \right) \right]. \quad (2.26)$$

Si ahora separamos la suma tenemos

$$A_k = \sqrt{m} \sqrt{\frac{1}{2(N+1)}} \Im \left[\sum_{i=0}^N u_i e^{\frac{ik2\pi j}{2(N+1)}} + \sum_{i=0}^N -u_i e^{\frac{(2(N+1)-i)k2\pi j}{2(N+1)}} \right]. \quad (2.27)$$

y con un cambio de variable en la segunda suma, puesto que el coeficiente $2(N+1) - i$ varía entre $N+2$ y $2(N+1)$ y teniendo en cuenta que $u_0 = 0$ y que $u_{N+1} = 0$, obtenemos

$$A_k = \sqrt{m} \sqrt{\frac{1}{2(N+1)}} \mathfrak{F} \left[\sum_{i=0}^N u_i e^{\frac{ik2\pi j}{2(N+1)}} + \sum_{i=N+1}^{2N+1} -u_{2(N+1)-i} e^{\frac{ik2\pi j}{2(N+1)}} \right]. \quad (2.28)$$

y si ahora creamos el nuevo vector us de longitud $2(N+1)$ y definido por

$$\begin{aligned} us_0 &= 0 \\ us_i &= u_i, \text{ para } i = 1 \text{ a } N \\ us_{N+1} &= 0 \\ us_{N+i} &= -u_{N+2-i}, \text{ para } i = 2 \text{ a } N+1. \end{aligned} \quad (2.29)$$

finalmente llegamos a la ecuación con la Transformada Discreta de Fourier, es decir

$$\begin{aligned} A_k &= \sqrt{m} \sqrt{\frac{1}{2(N+1)}} \mathfrak{F} \left[\sum_{i=0}^{2N+1} us_i e^{\frac{ik2\pi j}{2(N+1)}} \right] \\ &= \sqrt{m} \sqrt{\frac{1}{2(N+1)}} \mathfrak{F}[\mathcal{DF}(us)]. \end{aligned} \quad (2.30)$$

Aparte de esta fórmula que nos resultará útil para programarla en MATLAB, también vamos a desarrollar otra fórmula para expresar u_i que nos permitirá simplificar el término no lineal. Empecemos por escribir

$$\sin\left(\frac{ik\pi}{N+1}\right) = -j \frac{1}{2} \left(e^{\frac{ik\pi j}{N+1}} - e^{-\frac{ik\pi j}{N+1}} \right) \quad (2.31)$$

Con esta ecuación podemos transformar como se escribe u_i a partir de la definición en función de A_k y tendremos

$$\begin{aligned}
 u_i &= \frac{1}{\sqrt{m}} \sqrt{\frac{2}{N+1}} \sum_{k=1}^N A_k \sin\left(\frac{ik\pi}{N+1}\right) \\
 &= -j \frac{1}{\sqrt{m}} \sqrt{\frac{1}{2(N+1)}} \sum_{k=1}^N A_k \left(e^{\frac{ik\pi j}{N+1}} - e^{-\frac{ik\pi j}{N+1}} \right)
 \end{aligned} \tag{2.32}$$

En este punto también podemos extender la definición de A_k para valores negativos y es fácil comprobar que $A_k = -A_{-k}$ basándonos en la fórmula (2.23). Además, basándonos en esta misma fórmula también tenemos que $A(0) = 0$ y que $A(N+1) = A(-(N+1)) = 0$. Ahora podemos transformar la ecuación anterior sumando sobre k desde $-(N+1)$ hasta $N+1$ y obtendremos

$$u_i = -j \frac{1}{\sqrt{m}} \sqrt{\frac{1}{2(N+1)}} \sum_{k=-(N+1)}^{N+1} A_k e^{\frac{ik\pi j}{N+1}} \tag{2.33}$$

2.2 Resultados Gráficos

Para mostrar gráficamente como se comportan las soluciones podemos utilizar el programa elaborado para el problema FPU para mostrar la dinámica lineal. Basta con usar un valor nulo para el factor no lineal, es decir $\alpha = 0$, para que el programa resuelva las ecuaciones de movimiento y calcule la energía. Como hemos indicado en el código del programa en MATLAB esta opción se corresponde con el valor $\mathbf{q}=0$ que anula el componente no lineal y permite obtener las gráficas del problema lineal.

Si utilizamos la condición inicial del problema FPU para el problema lineal obtenemos las siguientes gráficas. Como podemos ver la energía se mantiene constante y se concentra en un único modo y en las diferencias se ve como la oscilación sigue la pauta inicial.

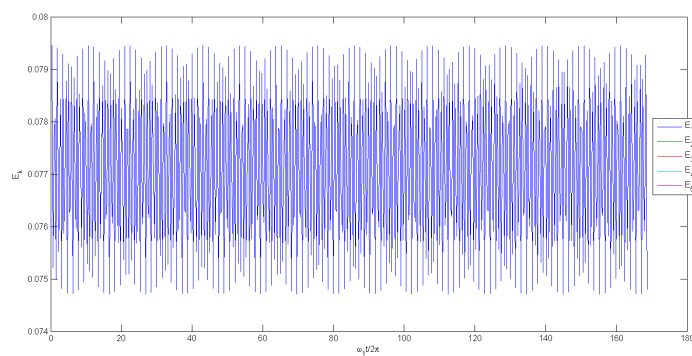


Figura 2.2: La energía se mantiene constante en el problema lineal

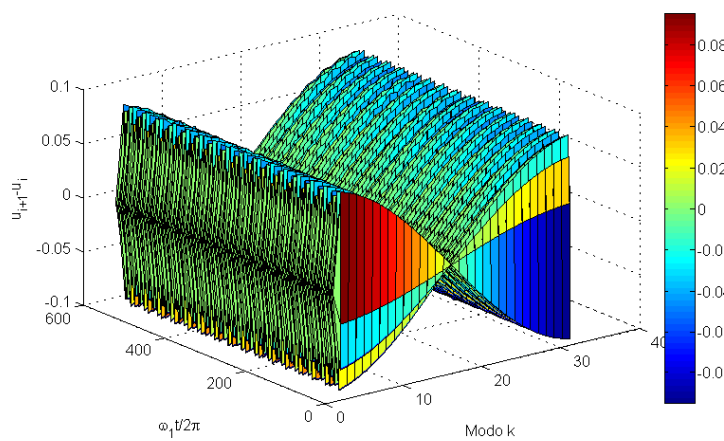


Figura 2.3: Las diferencias oscilan siguiendo la condición inicial

CAPÍTULO 3

Formulación del Problema de Fermi-Pasta-Ulam

3.1 Hamiltoniano de Fermi-Pasta-Ulam

Ahora que ya hemos encontrado la solución del problema lineal retomamos el problema de FPU. Básicamente el problema FPU parte del problema lineal y añade un término no lineal, en este caso cúbico, para alterar el hamiltoniano inicial. Además supondremos que tanto las masas como el coeficiente de Hooke son unitarios, es decir $m = 1$ y $\kappa = 1$, lo cual no hace desaparecer la generalidad del problema.

Con estas condiciones ($m = 1$, $\kappa = 1$ y término no lineal), la ecuación del hamiltoniano que hemos visto en el capítulo anterior (2.2) se escribirá (empezamos todos los sumatorios desde cero teniendo en cuenta que $p_0 = 0$ puesto que se supone que la primera partícula está anclada):

$$\mathcal{H} = \sum_{i=0}^N \frac{p_i^2}{2} + \sum_{i=0}^N \frac{1}{2} (u_{i+1} - u_i)^2 + \frac{\alpha}{3} \sum_{i=0}^N (u_{i+1} - u_i)^3, \quad (3.1)$$

con $\alpha \ll 1$ el coeficiente que mide la contribución no lineal al hamiltoniano. Vemos que hemos obtenido la misma ecuación que aparece en [4].

Con este hamiltoniano es fácil deducir las ecuaciones de movimiento correspondiente que serán:

$$\ddot{u}_i = [(u_{i+1} - u_i) - (u_i - u_{i-1})] + \alpha [(u_{i+1} - u_i)^2 - (u_i - u_{i-1})^2] \quad (3.2)$$

Al tener términos no lineales ya no puede obtenerse una solución analítica pero el procedimiento habitual es transformar la ecuación mediante el uso de los modos normales para estudiarla. Recordemos que los modos normales se asocian a los desplazamientos mediante la ecuación (2.21), es decir (teniendo en cuenta que $m = 1$)

$$A_k = \sqrt{\frac{2}{N+1}} \sum_{i=1}^N u_i \sin\left(\frac{ik\pi}{N+1}\right) \quad (3.3)$$

con las frecuencias de los autovalores presentes en (2.9), que con $m = 1$, $\kappa = 1$ y poniendo al cuadrado es igual a

$$\omega_k^2 = 4 \sin^2\left(\frac{k\pi}{2(N+1)}\right), \quad (3.4)$$

lo cual nos permite escribir el hamiltoniano como

$$\mathcal{H} = \frac{1}{2} \sum_{k=1}^N \left(\dot{A}_k^2 + \omega_k^2 A_k^2 \right) + \frac{\alpha}{3} \sum_{k,l,m=1}^N c_{klm} A_k A_l A_m \omega_k \omega_l \omega_m, \quad (3.5)$$

pudiendo determinar los coeficientes c_{klm} a partir de [15]. De hecho en [15] se explicita como calcular los coeficientes cuando el término no lineal es cúbico o de cualquier potencia superior, pero para ser completos en el desarrollo vamos a desarrollar el resultado utilizando nuestra formulación puesto que en el artículo no se muestran los detalles paso a paso.

3.1.1 Los factores no lineales

Lo primero que haremos para determinar los factores no lineales será usar la ecuación (2.33). Recordemos el resultado

$$u_i = -j \sqrt{\frac{1}{2(N+1)}} \sum_{k=-(N+1)}^{N+1} A_k e^{\frac{ik\pi j}{N+1}}. \quad (3.6)$$

Con este resultado vamos a calcular los valores de los términos no lineales. Estos términos se escriben como $\sum_i (u_{i+1} - u_i)^n$ siendo n el exponente del factor no lineal en el hamiltoniano, cuyo valor era 3 en el problema FPU pero pudiendo tener valores superiores también. Vamos a desarrollar una fórmula general para después aplicarla al potencial cúbico y explicitar así la interacción no lineal en el problema FPU. Escribamos ese término no lineal como $S_n = \sum_i (u_{i+1} - u_i)^n$, y calculamos esa suma como

$$\begin{aligned} S_n &= \sum_i \left(\left(-j \sqrt{\frac{1}{2(N+1)}} \right)^n \left[\sum_{k=-(N+1)}^{N+1} A_k \left(e^{\frac{(i+1)k\pi j}{N+1}} - e^{\frac{ik\pi j}{N+1}} \right) \right]^n \right) \\ &= \sum_i \left(\left(-j \sqrt{\frac{1}{2(N+1)}} \right)^n \left[\sum_{k=-(N+1)}^{N+1} A_k e^{\frac{ik\pi j}{N+1}} \left(e^{\frac{k\pi j}{N+1}} - 1 \right) \right]^n \right) \\ &= \sum_i \left(\left(\sqrt{\frac{1}{2(N+1)}} \right)^n \left[\sum_{k=-(N+1)}^{N+1} A_k e^{\frac{(i+1/2)k\pi j}{N+1}} (-j) \left(e^{\frac{k\pi j}{2(N+1)}} - e^{-\frac{k\pi j}{2(N+1)}} \right) \right]^n \right) \\ &= \sum_i \left(\left(\sqrt{\frac{1}{2(N+1)}} \right)^n \left[\sum_{k=-(N+1)}^{N+1} A_k e^{\frac{(i+1/2)k\pi j}{N+1}} 2 \sin \left(\frac{k\pi}{2(N+1)} \right) \right]^n \right) \\ &= \sum_i \left(\left(\sqrt{\frac{1}{2(N+1)}} \right)^n \left[\sum_{k=-(N+1)}^{N+1} A_k e^{\frac{(i+1/2)k\pi j}{N+1}} \omega_k \right]^n \right) \end{aligned} \quad (3.7)$$

Llegados a este punto tenemos que desarrollar la potencia en n . Normalmente el desarrollo de ese tipo fórmula se efectúa con el teorema multinomial que establece

$$(x_1 + x_2 + \dots + x_m)^n = \sum_{k_1+k_2+\dots+k_m=n} \binom{n}{k_1, k_2, \dots, k_m} \prod_{l=1}^m x_l^{k_l} \quad (3.8)$$

Pero en este caso esta fórmula no nos resulta útil porque no nos permite agrupar los términos de forma sencilla. Por este motivo vamos

a escribir la expansión de otra manera. Lo que vamos a necesitar es la expansión en función de los términos de la suma. Tendremos pues el resultado

$$(x_1 + x_2 + \cdots + x_m)^n = \sum_{k_1, k_2, \dots, k_n} x_{k_1} x_{k_2} \dots x_{k_n} \quad (3.9)$$

con cada k_l variando, de forma independiente, entre los valores 1 y m . Ahora con esta fórmula podemos expandir los términos y si partimos del punto anterior tendremos

$$\begin{aligned} S_n &= \sum_i \left(\left(\sqrt{\frac{1}{2(N+1)}} \right)^n \left[\sum_{k=-(N+1)}^{N+1} A_k e^{\frac{(i+1/2)k\pi j}{N+1}} \omega_k \right]^n \right) \\ &= \sum_i \left(\left(\sqrt{\frac{1}{2(N+1)}} \right)^n \left[\sum_{k_1, k_2, \dots, k_n} A_{k_1} A_{k_2} \dots A_{k_n} \omega_{k_1} \omega_{k_2} \dots \omega_{k_n} e^{\frac{(i+1/2)k_1\pi j}{N+1}} e^{\frac{(i+1/2)k_2\pi j}{N+1}} \dots e^{\frac{(i+1/2)k_n\pi j}{N+1}} \right] \right) \\ &= \sum_i \left(\left(\sqrt{\frac{1}{2(N+1)}} \right)^n \left[\sum_{k_1, k_2, \dots, k_n} A_{k_1} A_{k_2} \dots A_{k_n} \omega_{k_1} \omega_{k_2} \dots \omega_{k_n} e^{\frac{(i+1/2)(k_1+k_2+\dots+k_n)\pi j}{N+1}} \right] \right) \\ &= \left(\sqrt{\frac{1}{2(N+1)}} \right)^n \left[\sum_{i, k_1, k_2, \dots, k_n} A_{k_1} A_{k_2} \dots A_{k_n} \omega_{k_1} \omega_{k_2} \dots \omega_{k_n} e^{\frac{(i+1/2)(k_1+k_2+\dots+k_n)\pi j}{N+1}} \right] \\ &= \left(\sqrt{\frac{1}{2(N+1)}} \right)^n \left[\sum_{k_1, k_2, \dots, k_n} A_{k_1} A_{k_2} \dots A_{k_n} \omega_{k_1} \omega_{k_2} \dots \omega_{k_n} \sum_i \left(e^{\frac{(i+1/2)(k_1+k_2+\dots+k_n)\pi j}{N+1}} \right) \right] \end{aligned} \quad (3.10)$$

y puede verse que en el último paso hemos intercambiado el sumando sobre i con los sumandos sobre k_l puesto que son independientes. De esta manera podemos sumar sobre los términos en exponencial al tratarse de una suma geométrica. De esta manera tendremos

$$\sum_{i=0}^N \left(e^{\frac{(i+1/2)(k_1+k_2+\dots+k_n)\pi j}{N+1}} \right) = e^{\frac{(k_1+k_2+\dots+k_n)\pi j}{2(N+1)}} \sum_{i=0}^N \left(e^{\frac{i(k_1+k_2+\dots+k_n)\pi j}{N+1}} \right) \quad (3.11)$$

Si llamamos $(N+1)D(k_1 + k_2 + \cdots + k_n)$ a esta suma tendremos lo siguiente

$$\begin{aligned} D(0) &= 1 \\ D(\pm 4k(N+1)) &= 1 \\ D(\pm 2(2k+1)(N+1)) &= -1 \\ D() &= 0 \end{aligned} \quad (3.12)$$

Finalmente llegamos al valor de la suma

$$S_n = (N + 1) \left(\sqrt{\frac{1}{2(N+1)}} \right)^n \left[\sum_{k_1, k_2, \dots, k_n} A_{k_1} A_{k_2} \dots A_{k_n} \omega_{k_1} \omega_{k_2} \dots \omega_{k_n} D(k_1 + k_2 + \dots + k_n) \right] \quad (3.13)$$

Si ahora usamos este resultado para el problema original FPU con $n = 3$ y teniendo en cuenta que aquí la suma es de $-(N + 1)$ a $N + 1$ y que $A_{N+1} = A_{-(N+1)} = 0$, que $A_0 = 0$ y que $A_k \omega_k = A_{-k} \omega_{-k}$, podemos escribir los coeficientes c_{klm} de la siguiente manera

$$\begin{aligned} c_{klm} &= 2(N + 1) \left(\sqrt{\frac{1}{2(N + 1)}} \right)^3 D(k + l + m) \\ &= \frac{1}{\sqrt{2(N + 1)}} D(k + l + m) \end{aligned} \quad (3.14)$$

Resultados de este mismo tipo aparecen en [16], [17] y en [15], pero aquí hemos detallado los resultados y mostrado el paso a paso para facilitar la comprensión.

3.2 Equipartición de la Energía

Para un sistema lineal, puesto que tenemos modos normales, la distribución inicial de la energía de cada modo seguirá así de forma indefinida puesto que no hay intercambio de energía entre ellos. Fermi pensaba que cualquier término no lineal por débil que fuese generaría un intercambio de energía entre los modos y finalmente se obtendría la equipartición de la energía. Al no poder conseguir una solución analítica, Fermi, Pasta y Ulam usaron la simulación numérica para estudiar el problema.

El teorema de equipartición de la energía en la mecánica clásica estadística establece que el valor medio de cada término cuadrático del hamiltoniano es igual a $k_B T/2$, siendo T la temperatura del sistema. Este resultado se deriva usando los valores medios sobre el conjunto. Esto significa que

$$\left\langle \frac{1}{2m} \dot{A}_k^2 \right\rangle_{ens} = \left\langle \frac{1}{2} \omega_k^2 A_k^2 \right\rangle_{ens} = \frac{k_B T}{2}, \forall k \quad (3.15)$$

y por lo tanto

$$\langle E \rangle_{ens} = k_B T, \forall k. \quad (3.16)$$

La hipótesis de ergodicidad establece que la media sobre el conjunto $\langle \dots \rangle_{ens}$ y la media temporal $\langle \dots \rangle_{tiempo}$ son iguales y la media sobre el conjunto puede obtenerse efectuando una media temporal, calculando la evolución del sistema. Como sabemos la equipartición de la energía no ocurre para el problema lineal y el experimento de Fermi-Pasta-Ulam demostró que aunque al principio parecía que efectivamente la energía se repartía entre los modos al pasar el tiempo la energía volvía a concentrarse casi totalmente en el modo inicial, dando lugar a una recurrencia temporal (y también a una super recurrencia), contradiciendo la hipótesis de la equipartición de la energía para el problema no lineal como se pensaba originalmente.

3.3 Creación del algoritmo

Ahora que tenemos la ecuación del nuevo problema podemos crear un algoritmo en MATLAB para resolver el problema numéricamente, como hicieron Fermi, Pasta, Ulam y Tsingou en su experimento original. Gracias a las herramientas actuales la resolución computacional será bastante más sencilla que antaño.

Partiremos de la ecuación de movimiento resolviéndola numéricamente y a partir de ahí calcularemos la energía del sistema para ver como se reparte la energía inicial entre los modos. Puesto que usaremos MATLAB como lenguaje de programación construiremos el algoritmo para adaptarse a esta herramienta y sacar partido de sus posibilidades de resolución.

La herramienta principal de resolución de ecuaciones diferenciales es la función de MATLAB `ode45`. Esta función se usa para resolver sistemas de ecuaciones del tipo $\mathbf{y}' = f(t, \mathbf{y})$, siendo t la variable que

normalmente representa el tiempo, y siendo \mathbf{y} el vector de valores a resolver. En caso de tener una ecuación newtoniana del tipo $\ddot{u} = f(t, u)$ se construye un sistema de ecuaciones con $\dot{u} = v$ y $\dot{v} = f(t, u)$ y así se puede crear un sistema vectorial con $y_1 = v$ e $y_2 = u$ lo cual nos da que $\mathbf{y}' = f(t, \mathbf{y})$.

En este caso tenemos N incógnitas, que se corresponden con las posiciones u_i de las N partículas. Si transformamos esto en un sistema vectorial lo primero será tener en cuenta las condiciones iniciales. Si llamamos \mathbf{b} a las posiciones y las velocidades iniciales tendremos que

$$b_i = u_i(t = 0) = a \sin\left(\frac{\pi i}{N + 1}\right), \text{ para } i = 1 \text{ a } N \quad (3.17)$$

$$b_{N+i} = v_i(t = 0) = \dot{u}_i(t = 0) = 0, \text{ para } i = 1 \text{ a } N,$$

y esto quiere decir que en MATLAB las condiciones iniciales son

$$\mathbf{a}=1; \mathbf{b}(\mathbf{I})=\mathbf{a}*\sin(\pi*\mathbf{I}/(\mathbf{N}+1)); \mathbf{b}(\mathbf{I}+\mathbf{N})=0; \quad (3.18)$$

es decir las primeras N componentes del vector de valores iniciales son las posiciones iniciales y las componentes $N + 1$ a $2N$ son las velocidades iniciales. Aunque parezca extraño para el uso de la función `ode45` en MATLAB la función $f(t, \mathbf{y})$ usa el vector \mathbf{y} de forma inversa, es decir que internamente las componentes 1 a N son para las velocidades (como en el sistema explicado anteriormente cuando se ha convertido la ecuación diferencial de segundo grado en un sistema de primer grado) y las componentes $N + 1$ a $2N$ son para las posiciones.

Teniendo en cuenta esto para determinar las posiciones u_i de cada partícula nos queda por establecer el sistema de ecuaciones basándonos en la de movimiento establecida anteriormente. Es decir, para las primeras N componentes tendremos las velocidades que se corresponderán con los valores $N + 1$ a $2N$ del vector \mathbf{y} , y para las $N + 1$ a $2N$ tendremos la ecuación de movimiento de las posiciones, y por lo tanto si escribimos

$$\begin{aligned} y_i &= u_i, \text{ para } i = 1 \text{ a } N \\ y_{N+i} &= v_i = \dot{u}_i, \text{ para } i = 1 \text{ a } N \end{aligned} \quad (3.19)$$

y teniendo en cuenta que los extremos están inmóviles, es decir que $u_0 = \dot{u}_0 = 0$ y $u_{N+1} = \dot{u}_{N+1} = 0$, tendremos

$$\ddot{u}_i = (u_{i+1} + u_{i-1} - 2u_i) + \alpha [(u_{i+1} - u_i)^2 - (u_i - u_{i-1})^2]. \quad (3.20)$$

Lo cual en MATLAB si llamamos D al vector conjunto tendremos que el algoritmo se escribirá

$$\begin{aligned} D(I) &= y(N+I), \text{ para } I = 1 \text{ a } N \\ D(N+1) &= y(2) - 2*y(1) + \alpha * ((y(2) - y(1))^2 - (y(1) - y(0))^2) \\ D(N+I) &= y(I+1) + y(I-1) - 2*y(I) + \alpha * ((y(I+1) - y(I))^2 - (y(I) - y(I-1))^2), \text{ para } I = 2 \text{ a } N-1 \\ D(2*N) &= y(N-1) - 2*y(N) + \alpha * (y(N)^2 - (y(N) - y(N-1))^2). \end{aligned} \quad (3.21)$$

De esta manera tenemos definida la función que usaremos para resolver la ecuación de movimiento gracias a `ode45` en MATLAB, que nos devolverá tanto las posiciones (en las primeras N componentes) como las velocidades (en las componentes $N + 1$ a $2N$). Es decir que si llamamos $Y(IT, 1:N)$ al resultado de `ode45`, con IT el vector de tiempo, y definimos $YX(IT, J)$ como el vector de posiciones, y $YV(IT, J)$ como el vector de velocidades, y considerando que el primer punto es fijo, tendremos

$$\begin{aligned} YX(IT, 1:N+1) &= [0 \ Y(IT, 1:N)] \\ YV(IT, 1:N+1) &= [0 \ Y(IT, N+1:2*N)] \end{aligned} \quad (3.22)$$

Si queremos obtener también la derivada espacial de los desplazamientos (diferencias entre la partícula i e $i + 1$) a lo largo del tiempo crearemos una matriz de diferencias que en MATLAB denominaremos $DifY(IT, :)$ y cuyo cálculo será

$$DifY(IT, J) = Y(IT, J+1) - Y(IT, J), \text{ para } J = 2 \text{ a } N - 1; \quad (3.23)$$

Además de esto también tenemos que especificar las opciones de la función `ode45` lo cual se hace con la función `odeset`. En esta función especificaremos la tolerancia de la resolución numérica (con la opción `'Reltol'`), la manera de mostrar los resultados (con la opción

'OutputFcn') y qué componentes de la solución deben pasar a la función de salida (con la opción 'OutputSel'). Esto en MATLAB se escribirá

$$\text{options}=\text{odeset}(\text{'Reltol'},1\text{e-}4,\text{'OutputFcn'},\text{'odeplot'},\text{'OutputSel'},[1,2,N]); \quad (3.24)$$

Si queremos calcular la energía tendremos que calcular los coeficientes de los modos normales. Como hemos establecido anteriormente la fórmula para los coeficientes es

$$A_k = \sqrt{\frac{2}{N+1}} \sum_{i=1}^N u_i \sin\left(\frac{ik\pi}{N+1}\right) \quad (3.25)$$

que como hemos visto también puede escribirse mediante la parte imaginaria, \Im , de la Transformada Discreta de Fourier \mathcal{DF} como:

$$A_k = \sqrt{\frac{1}{2(N+1)}} \Im[\mathcal{DF}(us)], \quad (3.26)$$

con us definido por (recordemos que $u_0 = 0$ y $u_{N+1} = 0$)

$$\begin{aligned} us_0 &= 0 \\ us_i &= u_i, \text{ para } i = 1 \text{ a } N \\ us_{N+1} &= 0 \\ us_{N+i} &= -u_{N+2-i}, \text{ para } i = 2 \text{ a } N+1. \end{aligned} \quad (3.27)$$

Para calcular el valor de la parte imaginaria de la Transformada Discreta de Fourier de los desplazamientos u_i utilizaremos la Transformada Rápida de Fourier o *FFT* en MATLAB. Con esto podremos calcular tanto el coeficiente del modo normal A_k en función de los desplazamientos como su derivada \dot{A}_k en función de las velocidades. En MATLAB usaremos `sXF(IT, :)` para los coeficientes A_k y `sVF(IT, :)` para \dot{A}_k , con lo cual tendremos

$$\begin{aligned} \text{sXF(IT, :)} &= \text{imag}(\text{fft}([\text{YX(IT,1:N+1)} \ 0 \ -\text{YX(IT,N+1:-1:2)}]))/\text{sqrt}(2*(N+1)); \\ \text{sVF(IT, :)} &= \text{imag}(\text{fft}([\text{YV(IT,1:N+1)} \ 0 \ -\text{YV(IT,N+1:-1:2)}]))/\text{sqrt}(2*(N+1)); \end{aligned} \quad (3.28)$$

Con estos valores podremos resolver la calcular la energía de cada modo, definida por $E_k = \frac{1}{2} \left(\dot{A}_k^2 + \omega_k^2 A_k^2 \right)$ lo cual en MATLAB se escribirá como

$$\text{Energ}(\text{IT}, 1:N) = (\text{omegak2}(1:N) .* (\text{sXF}(\text{IT}, 2:N+1) .^2) + \text{sVF}(\text{IT}, 2:N+1) ^2) / 2. \quad (3.29)$$

Con todos estos resultados ya podemos crear el programa de resolución que dependiendo de las condiciones iniciales (FPU, Zabusky-Deem, Toda) nos permitirá responder a las preguntas planteadas como veremos en los siguientes capítulos.

CAPÍTULO 4

Desarrollos del Problema FPU

4.1 Los desarrollos iniciales a partir del problema FPU

En este capítulo expondremos los desarrollos que surgieron en los años posteriores a la publicación de los resultados del problema FPU. Tras la presentación de los resultados numéricos que contradecían lo esperado por Fermi el problema FPU se convirtió en un foco de interés para físicos y matemáticos que intentaban explicar los resultados obtenidos numéricamente. En el esquema que presentamos a continuación ([18]) vemos los principales artículos que trataban sobre los problema FPU o que tenían relación con él.

En la columna de la izquierda se ven los artículos principales del problema FPU que ya hemos mencionado:

- las simulaciones originales de 1954 publicadas de forma restringida en Los Álamos en 1955 pero cuyos resultados fueron circulando en el ínterin entre físicos y matemáticos y que después fueron publicadas abiertamente en 1965 ([1]),
- las simulaciones de mayor duración de 1961 publicadas internamente en Los Álamos en 1968 y postermiente de forma abierta en 1972 ([5]).

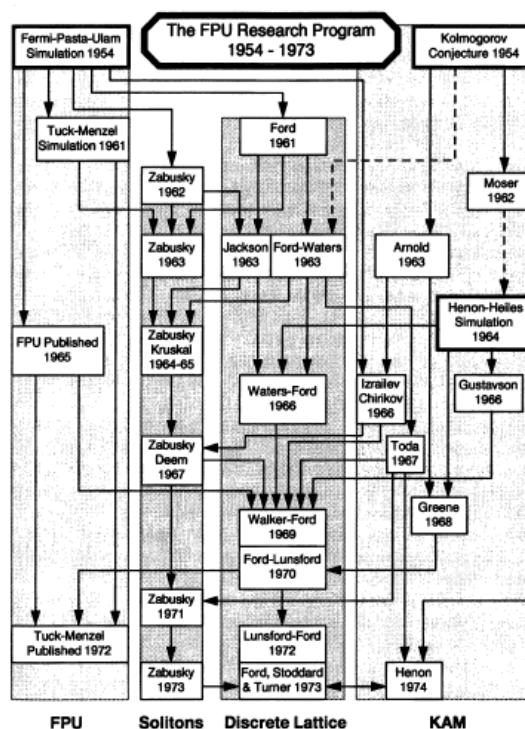


Figura 4.1: Cadena de osciladores

Como vemos en la imagen tras surgir los resultados originales y en paralelo a estos análisis surgieron varias vías de investigación. Las ramas principales que podemos reseñar en estos años posteriores a los resultados del problema FPU son:

- El descubrimiento de los solitones y la relación del problema FPU con la ecuación KdV por parte principalmente de Zabusky, con la colaboración de Kruskal, Deem y Tappert.
- La red discreta por parte principalmente de Ford, con la colaboración de Waters, Walker, Lunsford, Stoddard, Turner y las aportaciones de Jackson.
- La teoría de Kolmogorov-Arnold-Moser también conocida como KAM aplicada a los sistemas dinámicos que fue desarrollada a partir del trabajo de estos tres investigadores y que esclarece ciertos puntos del problema FPU mediante el uso de la teoría de la perturbación del hamiltoniano (y posteriormente de la teoría de la estabilidad débil de Nekhoroshev).

Basándonos en estas 3 áreas vamos a revisar los resultados y artículos principales de cada una de ellas en relación con el problema FPU y así estableceremos también la conexión con las simulaciones numéricas efectuadas como parte de este trabajo. Además también consideraremos la red de Toda que también fue parte de las simulaciones numéricas y cuyos artículos principales serán revisados.

4.2 La teoría de los solitones

Como hemos señalado anteriormente el descubrimiento de las ondas solitarias como soluciones del problema no lineal, que pasarían a llamarse solitones, se debe principalmente al trabajo de Norman J. Zabusky ([6], [19], [20]) y de sus colaboraciones con Kruskal primero ([21] y [7]), con Deem ([14]) y con Tappert ([22]). En este apartado vamos a presentar los resultados principales sobre solitones incluidos en estos artículos y como se relacionan con el problema FPU y con las simulaciones numéricas efectuadas.

El método empleado por Zabusky para analizar el problema fue utilizar el límite continuo, es decir en vez de usar el modelo discreto de partículas empleado en el problema FPU (lo cual se hizo para utilizar las posibilidades computacionales del MANIAC I que necesitaba un problema discreto para realizar los cálculos) se plantea el problema de una cuerda o hilo continuo que oscila con una componente no lineal.

En su primer artículo sobre el tema en 1962 ([6]), Zabusky calcula las soluciones exactas de las vibraciones de una cuerda continua no lineal. Las ecuaciones en derivadas parciales se obtienen haciendo que el número de partículas aumente al infinito ($N \rightarrow \infty$), el espacio entre partículas tiende hacia cero ($h \rightarrow 0$), el producto de estos dos valores permanece constante y finito ($L = Nh$) y el término no lineal (α/N) también es finito no nulo con valor $\frac{1}{2}\epsilon$. Para el problema FPU con un término cuadrático la ecuación de diferencias se convierte en

$$\frac{\partial^2 u}{\partial t^2} = \left(1 + \epsilon \frac{\partial u}{\partial x}\right) \frac{\partial^2 u}{\partial x^2}. \quad (4.1)$$

Mediante una transformación para invertir las variables dependientes e independientes, Zabusky consigue obtener un sistema de ecuaciones lineales que mediante el método de integración de Riemann permite obtener soluciones analíticas de las variables dependientes que después han de *desinvertirse* para obtener las posiciones y las velocidades. De esta manera se consigue mostrar que u_{xx} se vuelve singular, es decir que u_x tiene una discontinuidad. Pero en el problema FPU, que se corresponde con una discretización, no aparece esta discontinuidad por lo que el modelo lineal no describe correctamente el comportamiento del problema FPU y para ello hay que añadir términos que modelicen el carácter discreto lo cual se puede conseguir introduciendo derivadas superiores como u_{xxxx} y u_{ttt} .

Tras esta primera aproximación al problema en 1962, Zabusky lo retomó con Kruskal en 1964 y 1965 ([21] y [7]) introduciendo, como había mencionado en sus artículos anteriores, derivadas parciales de orden superior. Lo primero fue añadir un término de orden 4 para simular que el problema FPU era discreto. La ecuación se escribía entonces

$$\frac{u_{tt}}{c^2} = (1 + \epsilon u_x) u_{xx} + \left(\frac{h^2}{12}\right) u_{xxxx}, \quad (4.2)$$

con $c^2 = h^2 \omega_0^2$ la velocidad de onda en el límite lineal. Llegados a este punto Zabusky introdujo el cambio de variable

$$y = \frac{1}{2} \left(\frac{2}{3\epsilon} \left[(1 + \epsilon u_x)^{3/2} - 1 \right] - \frac{u_t}{c} \right), \quad (4.3)$$

que se trata de una cantidad no variable de la ecuación anterior. Con esta sustitución y poniéndose en unas coordenadas móviles ($x \rightarrow x - ct$), la ecuación se convierte en la ecuación de Korteweg-de Vries (KdV), es decir

$$y_\tau + yy_x + \delta^2 y_{xxx} = 0, \quad (4.4)$$

con $\tau = \epsilon ct/2$ y $\delta^2 = h^2/12\epsilon$. Analizando esta ecuación Zabusky y Kruskal descubrieron la existencia de ondas solitarias no dispersivas que bautizaron con el nombre de solitones. La característica principal de los solitones es que varios pueden coexistir simultáneamente en un mismo

medio y pueden propagarse los unos a través de los otros sin dispersarse, recomponiéndose tras atravesarse mutuamente. Así pues el solitón es una onda no lineal con las propiedades de una partícula. El solitón puede escribirse como

$$y = y_{\infty} + (y_0 - y_{\infty}) \operatorname{sech}^2 \left(\frac{x - x_0}{\Delta} \right) \quad (4.5)$$

con y_0 , y_{∞} y x_0 constantes arbitrarias y

$$\begin{aligned} \Delta &= \delta \left[\frac{y_0 - y_{\infty}}{12} \right]^{-\frac{1}{2}}, \\ c &= y_{\infty} + \frac{y_0 - y_{\infty}}{3}, \\ y &= Y(x - ct). \end{aligned} \quad (4.6)$$

La principal diferencia entre el problema FPU y el análisis de Zabusky era que en el problema FPU original las condiciones de frontera eran dos puntos fijos, uno a cada extremo, y sin embargo Zabusky utilizaba condiciones de frontera periódica para facilitar los cálculos y para que se propagasen los solitones. Esto concuerda con los resultados de nuestra simulación numérica para $q=4$ cuando se introducían los solitones sobre la red de Toda, con las ondas rebotando en las fronteras.

Además, Zabusky también realizó simulaciones numéricas con Deem ([14]) para analizar el comportamiento del modelo continuo versus el modelo discreto, y analizando como se comportaba el sistema dependiendo del tipo de excitación inicial, comprobado así que incluso si se excitaban los modos superiores no se alcanzaba la equipartición de la energía sino que con el paso del tiempo el sistema todavía mostraba correlación entre los desplazamientos de grupos de partículas en vez de un comportamiento aleatorio, como puede verse en la simulación numérica para $q=6$.

Posteriormente Zabusky publicaría artículos adicionales sobre el tema ([22] y [20]), profundizando en el análisis de los solitones en el problema continuo y en especial para la ecuación KdV.

4.3 La red discreta y la teoría de la perturbación

Antes de la popularización de la teoría KAM, la explicación del problema FPU tenía dos vertientes: la primera, formulada principalmente por Zabusky mediante la utilización de un modelo continuo, y la segunda centrada en la teoría de la perturbación, formulada por Ford en primer lugar ([10]) y que después sería seguida por Jackson ([23] y [16]) y Waters ([24] y [25]), y finalmente ampliada con la teoría KAM por el propio Ford con Walker ([26]) y Lunsford ([27] y [28]).

En el primer artículo de Ford ([10]), publicado en 1961, se utilizaba la teoría de perturbación pero sin el apoyo de la teoría KAM por lo que Ford sólo podía conjeturar que los diferentes modos comportían energía sólo si eran linealmente dependientes, y éste no era el caso puesto que los modos definidos por las frecuencias $\omega_k = 2 \sin(k\pi/(N + 1))$ son independientes.

Dos años después en 1963, Jackson publicó dos artículos ([23] y [16]) con un nuevo acercamiento a la teoría de la perturbación para el problema FPU. En estos artículos Jackson mostraba que la energía inicial de los modos inferiores tenía que propagarse primero a los modos intermedios antes de llegar a los superiores. Además utilizando el concepto del tiempo de recurrencia de Poincaré calculaba una aproximación sobre el tiempo necesario para que el sistema retorne al estado inicial que, en caso de considerar el problema FPU como un caso de acoplamiento fuerte, se acercaba a los resultados numéricos de la simulación FPU.

A los pocos meses, Ford en colaboración con Waters ([24]) presentó otro artículo con una continuación posterior en 1966 ([25]). En estos artículos Ford ya consideraba que los modos podían compartir energía a diferencia de en su primer acercamiento. Sin realmente utilizar los resultados de la teoría KAM estos artículos conseguían explicar el comportamiento de estos sistemas mediante la teoría de la perturbación dentro de una región de estabilidad.

4.4 La teoría de Kolmogorov-Arnold-Moser

La teoría de Kolmogorov-Arnold-Moser o teoría KAM es una teoría matemática sobre el comportamiento de sistemas dinámicos. Partiendo de una conjetura de Kolmogorov en 1954 ([29]), Moser primero en 1962 ([30] y [31]) y en paralelo Arnold en 1963 ([32] y [33]) demostraron el teorema que sería el punto de partida para lo que se denominaría la teoría KAM. Esta teoría, que después tendría otras dos aportaciones principales, una de Arnold en 1964 ([34]) con lo que se denominaría la difusión de Arnold, y otra de Nekhoroshev en 1971 ([35]) con su estimado (que también es sujeto de estudio en [36]), era la respuesta a un problema que llevaba prácticamente parado desde los trabajos de Poincaré a finales del siglo XIX ([37]). Básicamente en las ecuaciones de la mecánica clásica o dinámica una vez vamos más allá del problema de los dos cuerpos y pasamos al problema de tres o más cuerpos las ecuaciones de movimiento no pueden resolverse de forma analítica. Es por ello que la técnica de resolución consiste en usar la teoría de la perturbación expandiendo las funciones de fuerza alrededor de un parámetro ϵ . El primer término es la parte lineal y puede resolverse pero a medida que se añaden términos el problema se vuelve más complejo y adicionalmente al truncar la expansión se pierde precisión a medida que pasa el tiempo.

No fue hasta 1954 cuando Kolmogorov postuló su conjetura ([29]), que presentaría en un congreso en 1957 ([38]), afirmando que la mayoría de toros periódicos (trayectorias confinadas a un toro en el espacio de fases) permanecían estables tras añadir una pequeña perturbación al hamiltoniano integrable, obteniendo trayectorias cuasi periódicas. Esta conjetura fue probada posteriormente por Moser y Arnold de forma independiente. El primer uso directo de estos resultados para realizar un estudio numérico fue llevada a cabo por Hénon y Heiles ([39]) para sus cálculos sobre la tercera integral del movimiento galáctico, que después fueron utilizados por Gustavson ([40]). Desde el punto de vista matemático numerosos cálculos numéricos fueron llevados a cabo por Greene ([41]) para estimar los límites de estabilidad de los mapas presentados por Moser en su demostración del teorema.

Llegados a este punto podemos preguntarnos por la relación entre la teoría KAM y el problema FPU. El primer trabajo que relacionó el problema FPU con la teoría KAM fue el artículo de Izrailev y Chirikov en 1966 ([12]). En ese trabajo Izrailev y Chirikov daban por hecho que la teoría KAM era aplicable al problema FPU, siendo el término en α equivalente a una perturbación del hamiltoniano, sin por ello demostrar que se daban las condiciones necesarias. Su objetivo era encontrar el punto a partir del cual el sistema ya no se comportaría como un sistema KAM sino que el sistema FPU se convertiría en ergódico. Si tomamos por bueno el hecho de que el problema FPU cumple con las condiciones de la teoría KAM (de hecho no existe todavía una prueba matemática completa de este hecho) entonces es de suponer que siendo la perturbación pequeña el sistema FPU tendrá un comportamiento cuasi-periódico y por lo tanto existirá un periodo de recurrencia con respecto a los modos lineales. De ahí que la teoría KAM permita explicar de forma intuitiva los resultados del problema FPU.

4.5 La red de Toda

Aunque la red de Toda que expuso Morikazu Toda en 1967 ([11]) no es en sí una teoría física como la teoría KAM o como los solitones sino simplemente un tipo de interacción que permite resolver las ecuaciones de movimiento, el hecho de que puedan integrarse las ecuaciones de forma analítica hace que sea muy interesante estudiarla porque permite obtener soluciones exactas y estudiar su comportamiento de forma sencilla. En este problema Toda estudiaba un sistema de una cadena de partículas con una interacción no lineal (exponencial como se verá a continuación) en vez de con un término cuadrático o cúbico. La ecuación de movimiento de este sistema se escribe

$$m\ddot{u}_i = -\frac{d\phi}{dr}(u_i - u_{i-1}) + \frac{d\phi}{dr}(u_{i+1} - u_i), \quad (4.7)$$

con r el desplazamiento entre dos partículas, es decir que por definición tenemos $r_i = u_i - u_{i-1}$, y $\phi(r)$ es el potencial de interacción (equivalente al coeficiente del muelle en el caso FPU) definido por Toda para que sea

integrable, y definida por

$$\phi(r) = \frac{a}{b}e^{-br} + ar, \text{ con } a, b > 0, \quad (4.8)$$

con la condición inicial $d\phi/dr(0) = 0$.

Con este potencial se pueden obtener soluciones analíticas del problema y eso hace que este problema constituya una manera accesible de estudiar estructuras no lineales del tipo FPU con soluciones en forma de solitones calculadas tanto de forma numérica como analíticas. Véase el libro de Toda, [13], para una referencia completa sobre el problema. Además en el límite continuo también puede describirse por la ecuación KdV como el problema FPU. Es por ello que esta misma red fue estudiada por Ford, Stoddard y Turner ([42]) y por Henon ([43]) desde el punto de vista de su integrabilidad.

Ahora que hemos repasado los principales desarrollos iniciales tras conocerse el problema FPU y como se relacionan con otras disciplinas de la Física como la teoría KAM, vamos a pasar a las simulaciones numéricas del problema mediante el programa en MATLAB.

CAPÍTULO 5

Resultados de la Simulación del Problema FPU

Como presentamos al principio, la resolución numérica del problema se hará usando como punto de partida un programa en MATLAB presente en [4] que ha sido ampliado para responder a las preguntas planteadas anteriormente mediante la creación de gráficas y animaciones. Este programa, `fpu.m`, junto con una función asociada, `fpu1.m`, contiene el código de todas las ejecuciones con una variable interna (`q`) que permite elegir entre las ejecuciones para cada una de las preguntas. De esta manera se simplifica su manejo y se facilita su presentación.

Además en la versión antigua era necesario especificar los parámetros tanto en el programa principal como en la función asociada lo cual podía causar errores en caso de discrepancia. Para que esto no sucediese se ha modificado el programa y la función para que los parámetros se conviertan en variables que puedan pasarse a la función dinámicamente desde el programa principal y así evitar errores. Dicho esto, podemos pasar a presentar cual es el objetivo del programa y a analizar como está estructurado para llevar a cabo los cálculos.

5.1 El programa en Matlab

5.1.1 Objetivo del programa

Para entender qué cálculos realiza el programa en MATLAB lo más sencillo es revisar los comentarios incluidos en éste, que explican para qué se creó. Veamos estos comentarios y las declaraciones de variables iniciales.

```
%-----%
% Main code for carrying out the numerical experiment
% Based on the code provided by Thierry Dauxois in
% http://perso.ens-lyon.fr/thierry.dauxois/fpu.html
% This code will solve the differential equation for the FPU problem
% Based on the value of q the code will resolve different questions
%-----%

% Clear workspace
clear

% Activation of variables based on the question
q = 6;

% Initial variables from original FPU problem
N=32;           % Number of particles must be a power of 2
alpha=0.25;     % Non linear parameter
TMAX=10800;     % Lattice time for FPU recurrence
amax=1;        % Amplitude
DT=20;         % Time Step (delta t)

% Variables for the different scenarios to be test
% q = 0 - we remove the non linear term
% q = 1 - original FPU simulation
% q = 2 - superrecurrence
% q = 3 - increasing amplitude to observe energy equipartition
% q = 4 - Toda lattice
% q = 5 - Recurrence period follows  $N^{(5/2)}$ 
% q = 6 - Zabusky-Deem initial condition
```

Como explican estos comentarios, el programa realiza la resolución numérica de las ecuaciones del problema de Fermi-Pasta-Ulam para un sistema de N masas interactuando de forma lineal más una perturbación no lineal de parámetro α . En la cuestión cero ($q=0$) resolvemos el problema puramente lineal puesto que el coeficiente α es nulo. En este caso será fácil ver como la energía se mantiene en el modo de la condición inicial (resultado que en este caso puede comprobarse analíticamente).

5.1.2 Estructura del programa

Dejando de lado el problema lineal, nos interesaremos principalmente por el programa no lineal. Para simular los experimentos computacionales de Fermi, Pasta y Ulam (y Tsingou) que dieron pie al problema FPU (esto se corresponde con la cuestión 1 es decir $q=1$), la resolución se hace en varias etapas, usando esta misma metodología para las demás cuestiones:

- Primero, el programa define los diferentes modos de frecuencia, ω_k^2 , que se corresponden con los modos del problema lineal antes de ser perturbado; además se definen las condiciones iniciales, normalizando la amplitud a la unidad, $a = 1$, anclando los extremos como condiciones de frontera y definiendo los valores iniciales, b_i , para las partículas o masas que definen el sistema y que para el experimento FPU se corresponde con una onda sinusoidal del primer modo
- Segundo, el programa resuelve la ecuación diferencial no lineal con las condiciones iniciales especificadas utilizando la función de MATLAB `ode45` diseñada para este tipo de problemas
- Tercero, con la solución de la ecuación diferencial calcularemos la transformada discreta de Fourier (función `fft` en MATLAB) y tomaremos la parte imaginaria (función `imag` en MATLAB) que se corresponde con los modos sinusoidales (la parte real serían los cosenos).
- Cuarto, al tener ya las posiciones calculamos la energía y la diferencias en desplazamientos
- Finalmente, ya podemos generar las gráficas para los modos de la energía

5.1.3 Parámetros del programa

Para hacer más clara la lectura del programa hemos incluido en los comentarios las descripciones de los parámetros utilizados. Los parámetros iniciales son

```
% Initial variables from original FPU problem
N=32;           % Number of particles must be a power of 2
alpha=0.25;    % Non linear parameter
TMAX=10800;    % Lattice time for FPU recurrence
amax=1;        % Amplitude
DT=20;         % Time Step (delta t)
```

Como se puede ver, los parámetros iniciales del programa son:

- **N**: el número de osciladores, que en este caso ha de ser una potencia de dos
- **alpha**: el coeficiente de no linealidad
- **TMAX**: tiempo de simulación
- **DT**: paso temporal (delta t)
- **amax**: amplitud, por defecto igual a la unidad

Estos parámetros son los elementos principales con los que jugaremos para obtener diferentes resultados. Con el coeficiente de no linealidad separamos el comportamiento lineal del no lineal, con el número de osciladores podemos estudiar el comportamiento de diferentes cadenas y su tiempo de recurrencia, con el tiempo de simulación podemos buscar la recurrencia y la super recurrencia, y con la amplitud podemos ver como para valores altos desaparece la recurrencia y se obtiene un estado de equipartición de la energía. Además tenemos el parámetro q que usamos para simular diferentes modelos y fijar diferentes valores para estos parámetros principales.

5.2 Ejecución del programa

5.2.1 Tipos de ejecución

Como vimos anteriormente en el apartado sobre los parámetros de entrada del programa, cinco son los parámetros sobre los que jugaremos para obtener los resultados. Esos parámetros son:

- el número de osciladores N , que debe ser una potencia de 2
- el coeficiente de no linealidad que en el experimento FPU era igual a 0,25 de ahí que usemos ese valor por defecto
- el tiempo máximo de ejecución para constatar la aparición de la recurrencia y de la super recurrencia
- el paso delta de tiempo para la simulación
- la amplitud que por defecto normalizamos con valor unidad y que al aumentar hace desaparecer la recurrencia

Estos cinco parámetros tienen como valor por defecto los valores que se usan para simular el experimento FPU. Como ya mencionamos al principio de la memoria se efectuarán simulaciones para intentar responder a varias preguntas presentes en [4] más otras que hemos añadido como el problema lineal o la superrecurrencia. Cada una de estas preguntas se corresponderá a un valor de la variable q que se utilizará para ejecutar el código necesario para cada pregunta y también para modificar los valores de los cinco parámetros mencionados anteriormente.

Vamos a volver a listar las preguntas que simularemos y los valores que utilizaremos para los parámetros. Básicamente, las ejecuciones serán las siguientes

- $q=0$. Antes de empezar, simularemos el problema lineal. En este caso con $q=0$ se especificará un valor $\alpha = 0$, dejando los demás parámetros intactos. Con este valor el modo inicial sinusoidal que se ha usado debería conservarse a lo largo de la ejecución como queda reflejado en la solución analítica que puede obtenerse para este problema

- q=1. Para $q=1$ simularemos el problema FPU, con los valores que se presentaban en el experimento original de [1]. Esto quiere decir que tendremos 32 osciladores, una amplitud de 1, un parámetro de no linealidad de 0,25 y que tanto el tiempo de ejecución y el paso están ajustados para que aparezca la primera recurrencia. Para este punto se generarán gráficos adicionales para que se puedan ver los solitones en movimiento.
- q=2. Para $q=2$ simularemos el problema FPU pero sólo para 16 osciladores ($N=16$) y para un tiempo superior para así ver la super recurrencia que se investigó por primera vez numéricamente por Tuck y Menzel en 1972 ([5]).
- q=3. Para $q=3$ aumentaremos la amplitud subiendo de 1 a 10, para comprobar como para valores altos de la amplitud no aparece la recurrencia y la energía se reparte entre los modos dándose la termalización.
- q=4. Para $q=4$ usaremos como condición inicial los solitones con *kink* característicos de la red de Toda para ver cómo se comporta el sistema de FPU. Para obtener resultados numéricos elegiremos una cadena de 128 osciladores y modificaremos el tiempo de ejecución.
- q=5. Para $q=5$ intentaremos demostrar que el período de recurrencia se comporta como $N^{5/2}$. Para comprobar este resultado realizaremos simulaciones variando el número de osciladores de 8 a 16, a 32, a 64, a 128 y hasta 256, modificando para cada uno el tiempo de ejecución y el paso temporal. Con estas simulaciones encontraremos los tiempos de recurrencia y presentaremos los resultados en un gráfico logarítmico comprobando que efectivamente el período de recurrencia se comporta como $N^{5/2}$.
- q=6. Para $q=6$ usaremos la condición inicial de Zabusky-Deem ([14]) y que corresponde a excitar el modo superior en vez del primer modo. En este caso se esperaba que se observaría la equipartición de la energía pero como ya se vio en el artículo de Zabusky y Deem en realidad se ven comportamientos con estructura en vez de una simple equipartición de la energía. Esto lo podremos ver gráficamente en la ejecución.

5.3 Presentación de los resultados

En esta sección vamos a incluir los resultados de las simulaciones para las diferentes cuestiones (valores de q) que hemos relatado anteriormente. Como hemos visto anteriormente para el problema lineal correspondiente a $q=0$ las gráficas ya fueron incluidas en la memoria, por lo que empezaremos directamente con el problema FPU, es decir $q=1$.

5.3.1 El Problema FPU ($q=1$)

Mostramos a continuación las gráficas de los resultados para el problema FPU inicial que se corresponden con el parámetro $q=1$. En este caso las condiciones iniciales concordan con las del problema FPU original, es decir que utilizaremos $N = 32$ partículas, con un coeficiente de no linealidad $\alpha = 0,25$, con una amplitud inicial $a = 1$ y con una condición inicial $b_i = \sin(i\pi/(N + 1))$ que se corresponde con el primer modo ($k = 1$). Lo primero que vamos a mostrar es el gráfico de la energía, E_k de los diferentes modos.

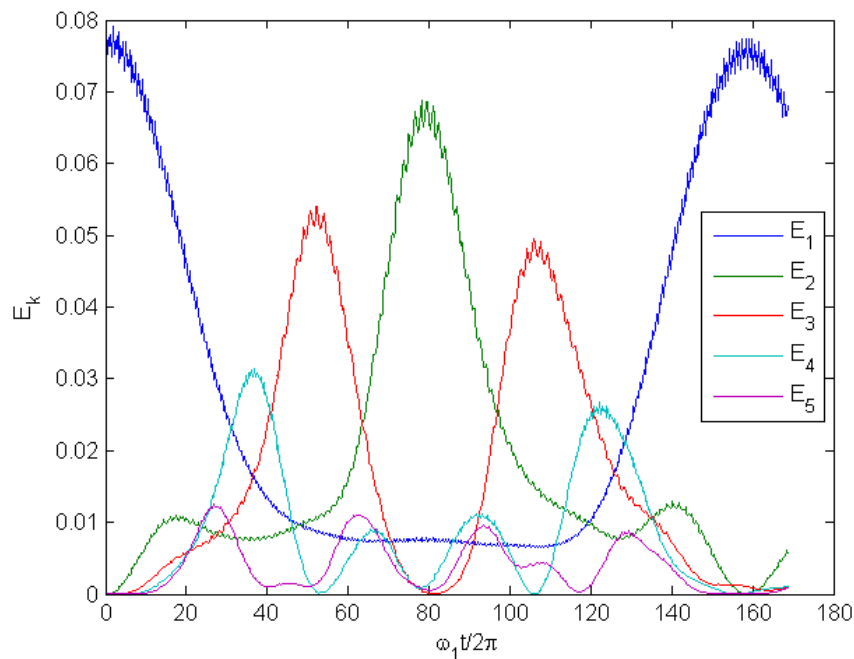


Figura 5.1: En el problema FPU la energía se reparte entre los modos y después vuelve al modo inicial

Esta figura se corresponde exactamente con la gráfica del problema FPU ([4]). Como podemos apreciar al cabo de 157 o 158 pasos la energía vuelve prácticamente en su totalidad al primer modo (en torno al 97 %). Esto es exactamente lo que descubrieron Fermi, Pasta y Ulam cuando realizaron su experimento numérico ([44]). Se aprecia claramente la recurrencia y como no se obtiene la equipartición de la energía.

También podemos mostrar la gráfica de las diferencias espaciales.

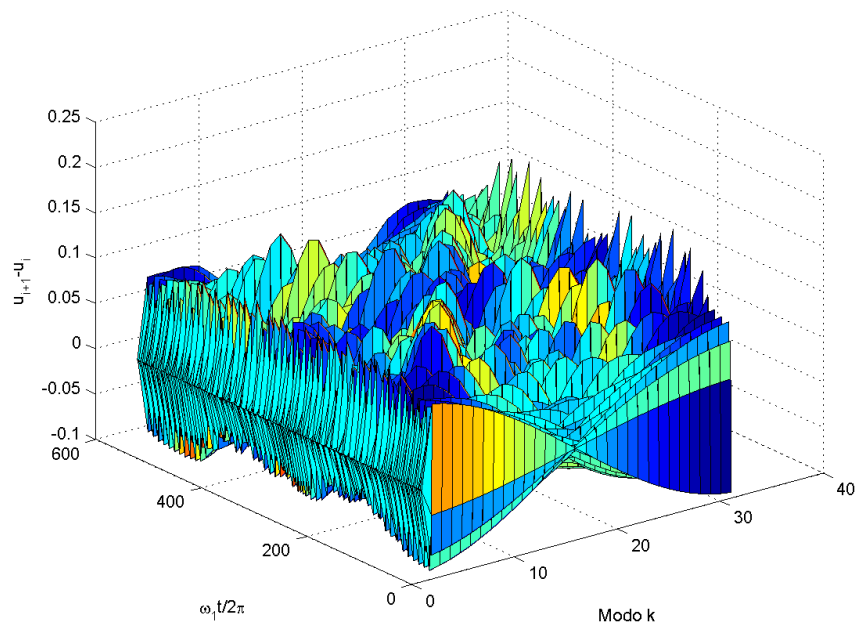


Figura 5.2: En las diferencias espaciales se aprecia la aparición de solitones en el problema FPU

En esta gráfica se aprecia como parece que una onda arranca a cada lado de los modos, y que se encuentran a medio camino y después reaparecen tras interactuar. Si queremos apreciar mejor los solitones podemos extraer únicamente los picos lo cual nos permite obtener una gráfica con mayor claridad, como vemos a continuación.

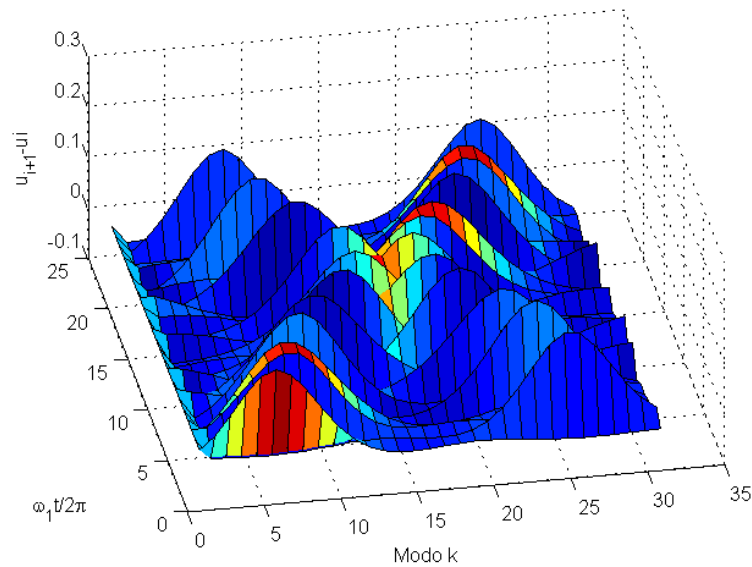


Figura 5.3: Extracción de los solitones en el problema FPU

Adicionalmente se puede también crear un vídeo con los solitones en movimiento. A continuación podemos verlo haciendo click sobre el borde la imagen y lanzando la aplicación de vídeo. Se aprecia con claridad como las ondas arrancan en extremos opuestos, se cruzan y después se recomponen.

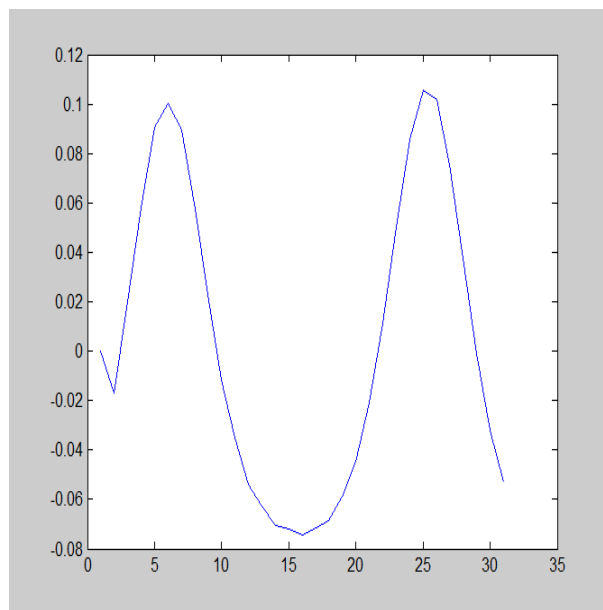


Figura 5.4: Vídeo de los solitones en el problema FPU

5.3.2 La super recurrencia en el Problema FPU ($q=2$)

Como se expuso anteriormente además de la recurrencia que se descubrió originalmente en el problema FPU, simulaciones numéricas más largas efectuadas posteriormente, como se expone en [5], mostraron que también existía una recurrencia de período más largo que se denominó super recurrencia. A continuación mostramos el resultado gráfico de la simulación numérica donde se aprecia este fenómeno. Las condiciones iniciales y los parámetros son los mismos que anteriormente.

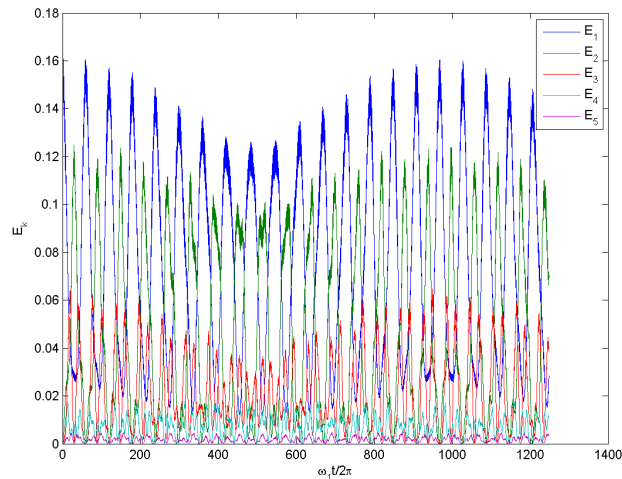


Figura 5.5: Aparición de la super recurrencia en el problema FPU

Como vemos en la gráfica se observa la super recurrencia a los 969 pasos y recuperando prácticamente toda la energía original, superando el 98% . Esto se corresponde con los resultados de Tuck y Menzel presentados en [5] que obtuvieron una gráfica semejante.

5.3.3 Aumento de la amplitud inicial en el problema FPU ($q=3$)

Para este problema vamos a aumentar la amplitud de la condición inicial pasando de una amplitud de $a = 1$ hasta $a = 10$. Cuando la amplitud aumenta la energía se reparte entre los modos y se alcanza la equipartición de la energía sin recurrencias del tipo FPU (ni super recurrencias). Ahora mostraremos las gráficas de la energía para valores crecientes de a para comprobar como se alcanza el equilibrio. Aquí vemos las gráficas.

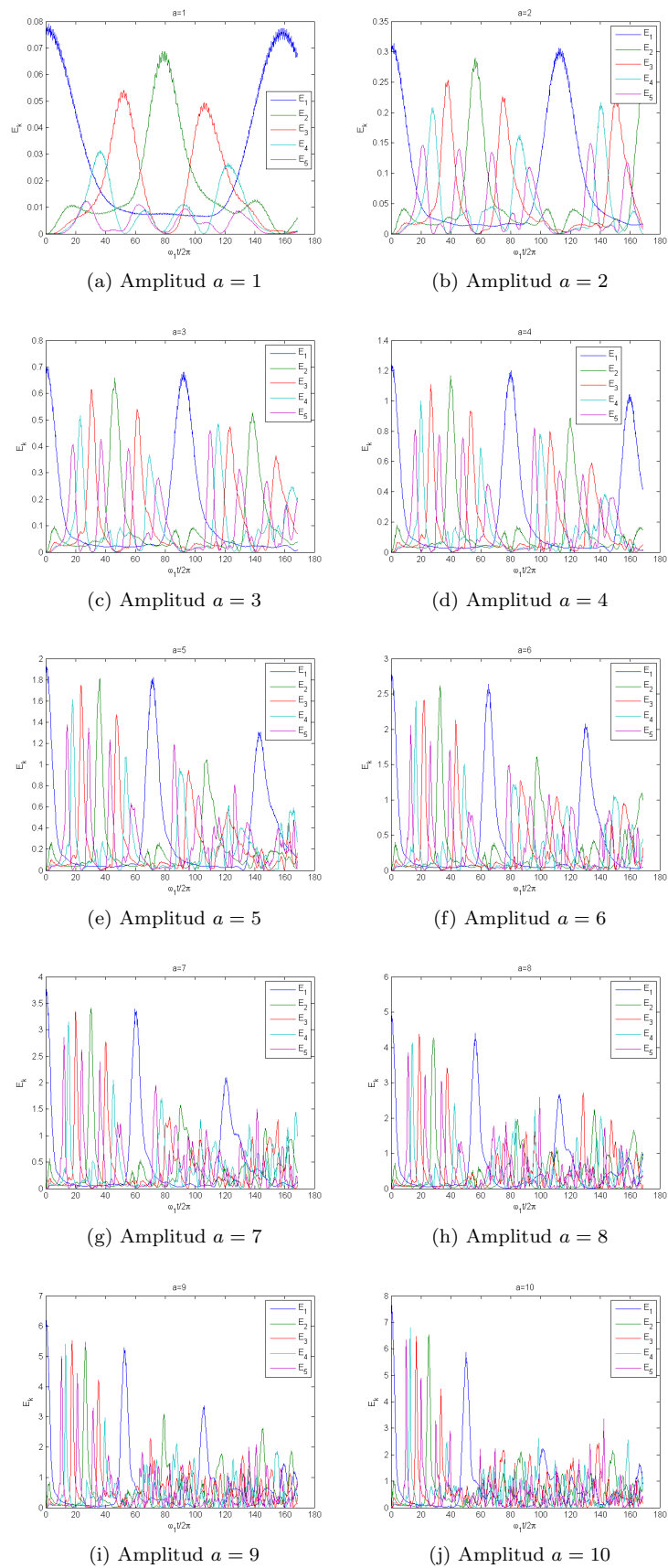


Figura 5.6: Energía de los modos dependiendo de la amplitud

Como se puede comprobar en la evolución de las gráficas con respecto a la amplitud a medida que ésta aumenta la recurrencia se diluye hasta que para $a = 10$ ya no se aprecia que el modo inicial recupere casi la totalidad de la energía sino que se reparte entre varios modos, mostrando así que se va hacia la equipartición de la energía.

5.3.4 La red de Toda (q=4)

En este problema utilizamos como condición inicial los solitones con *kink* de la red de Toda. En este caso se pueden conseguir resultados analíticos y la simulación numérica permite comprobar que efectivamente el sistema se comporta así. Básicamente se ve en la gráfica de diferencias de las posiciones como los ondas de solitones se propagan y rebotan al llegar al modo superior o inferior.

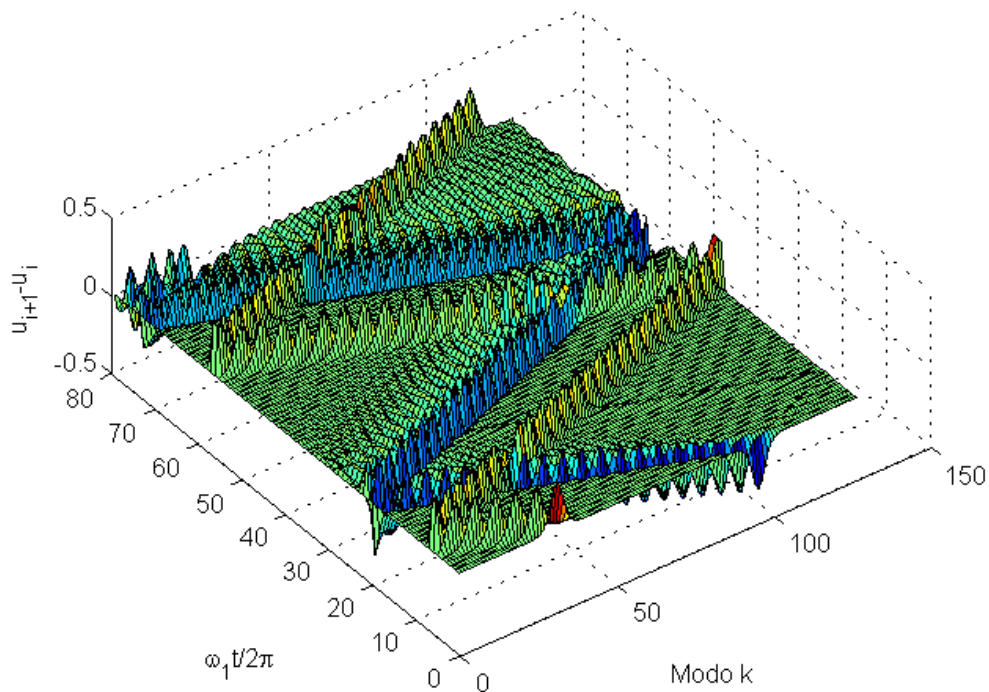


Figura 5.7: Los solitones en la red de Toda

5.3.5 El período de recurrencia (q=5)

Para estudiar el período de recurrencia vamos a hacer simulaciones cambiando el número de partículas, partiendo de $N = 8$, pasando por $N = 16$, $N = 32$, $N = 64$, $N = 128$ y finalmente $N = 256$. Puesto que sabemos que el período de recurrencia se comporta como $N^{5/2}$, hemos creado una gráfica logarítmica con los valores del período en función de $\omega_1 t / 2\pi$. Como hemos explicado anteriormente la fórmula presente en [4] es incorrecta, puesto que se dice que con $T_1 = 2N$

$$T_R = \frac{3}{\pi^{3/2}\sqrt{2}} \frac{N^{3/2}}{\sqrt{a\alpha}} T_1 \simeq 0,38 \frac{N^{5/2}}{\sqrt{a\alpha}}, \quad (5.1)$$

siendo a la amplitud y α la constante del término no lineal. Pero si revisamos esta fórmula vemos que con $T_1 = 2N$ el tiempo de recurrencia es

$$\begin{aligned} T_R &= \frac{3}{\pi^{3/2}\sqrt{2}} \frac{N^{3/2}}{\sqrt{a\alpha}} T_1 \\ &= \frac{3\sqrt{2}}{\pi^{3/2}} \frac{N^{5/2}}{\sqrt{a\alpha}} \\ &\simeq 0,76 \frac{N^{5/2}}{\sqrt{a\alpha}}. \end{aligned} \quad (5.2)$$

es decir que multiplicamos por 0,76 en vez de por 0,38.

Puesto que en las simulaciones numéricas usamos como medida temporal $\omega_1 t / 2\pi$, si ponemos el tiempo de recurrencia calculado anteriormente en esta escala y teniendo en cuenta que $\omega_1 = 2 \sin(\pi/2(N + 1))$ y llamamos τ_R a este período de recurrencia tendremos

$$\begin{aligned} \tau_R &= \frac{\omega_1 T_R}{2\pi} \\ &= \frac{3\sqrt{2}}{\pi^{5/2}} \sin\left(\frac{\pi}{2(N + 1)}\right) \frac{N^{5/2}}{\sqrt{a\alpha}}. \end{aligned} \quad (5.3)$$

Ahora podemos construir una tabla con los resultados esperados para los valores numéricos utilizados en nuestras simulaciones, con $a = 1$ y $\alpha = 0,25$ para los diferentes valores de N , lo cual nos dará

N	T_R	τ_R
8	276	15
16	1560	45
32	8827	133
64	49933	384
128	282466	1094
256	1597870	3108

Tabla 5.1: Tiempo de recurrencia analítico con $a = 1$ y $\alpha = 0,25$

Ahora presentaremos las gráficas obtenidas para los diferentes valores de N en el programa de MATLAB. Tenemos lo siguiente

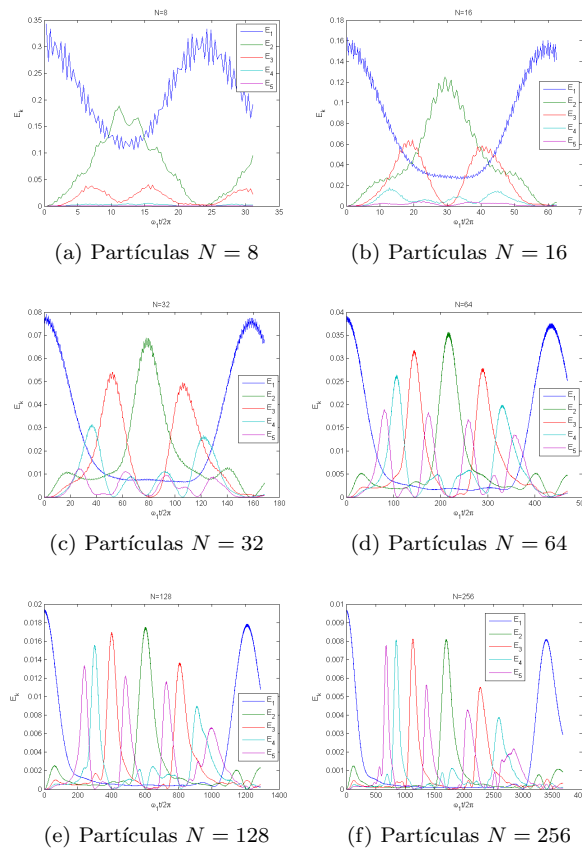


Figura 5.8: Tiempos de recurrencia para diferentes valores de N

Con estas gráficas podemos recrear la tabla anterior pero con los valores numéricos.

Tendremos en forma de tabla

N	T_R	τ_R
8	434	24
16	2043	60
32	10432	158
64	56425	434
128	312188	1210
256	1744527	3394

Tabla 5.2: Tiempo de recurrencia numérico con $a = 1$ y $\alpha = 0,25$

Es importante recalcar que los valores analíticos se calculan con la hipótesis del límite continuo y por lo tanto cuanto mayor es el número de partículas, es decir cuanto más crece N , más nos acercamos a ese límite continuo y más certera será la fórmula para el período de recurrencia.

Si representamos estos datos de forma logarítmica apreciamos como los valores analíticos se van aproximando a los valores numéricos. Veamos la gráfica.

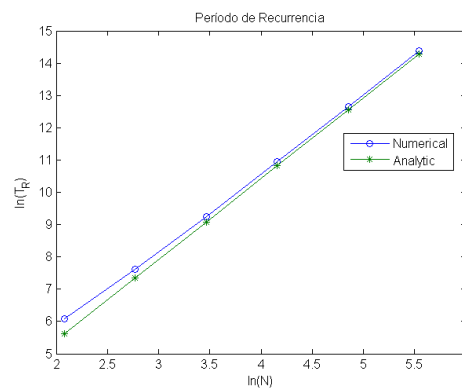


Figura 5.9: Estudio logarítmico del período de recurrencia

Como se aprecia en la gráfica logarítmica el período de recurrencia analítico se va acercando cada vez más al período de recurrencia numérico a medida que aumenta N , y además se ve que la pendiente también es equivalente. Así pues la fórmula de cálculo para el período de recurrencia nos permite obtener una aproximación cada vez mejor del valor a medida que N crece.

5.3.6 Condición inicial de Zabusky-Deem ($q=6$)

En este caso estudiaremos el comportamiento del sistema para la condición inicial de Zabusky-Deem que se corresponde con la excitación del modo superior en vez del primer modo, es decir que $b_i = \sin(Ni\pi/(N+1))$. El resultado esperado era la equipartición de la energía pero como veremos en la gráfica de diferencias se observa un comportamiento con estructuras. Veamos el resultado de la ejecución.

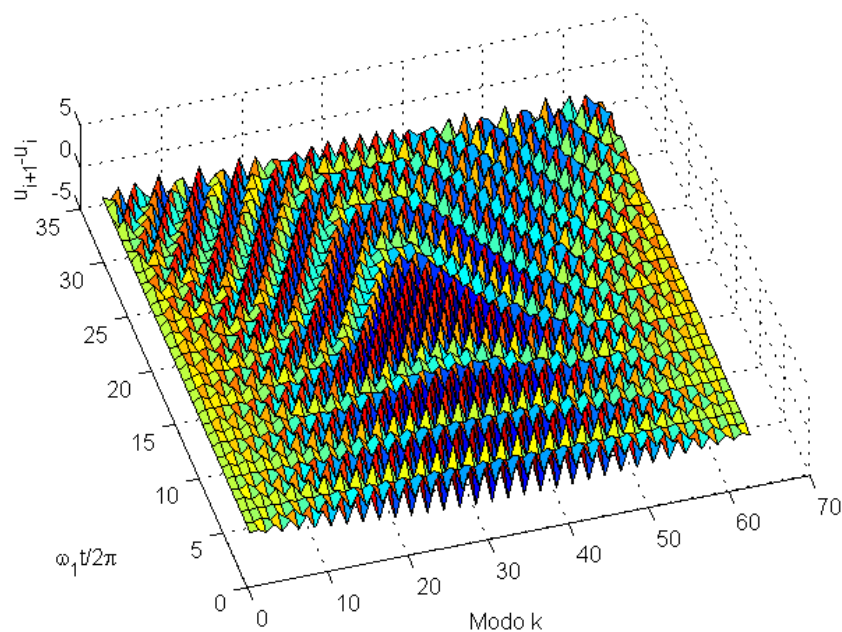


Figura 5.10: Estructuras con la condición inicial de Zabusky-Deem

5.3.7 Conclusiones

Como se ha podido ver con los resultados obtenidos, podemos extraer varias conclusiones para los casos estudiados

- En el problema FPU se comprueba claramente la aparición de solitones introducidos por Zabusky y Kruskal ([7])
- La recurrencia aparece con claridad y es proporcional a $N^{5/2}$, con un período de super recurrencia posterior ([5])
- Para amplitudes muy altas con respecto al término no lineal la recurrencia desaparece y nos acercamos a la equipartición de la energía que esperaba Fermi originalmente ([12])
- Para otro tipo de condiciones iniciales, como la Zabusky-Deem ([14]), el comportamiento no es de equipartición pero tampoco se obtiene una recurrencia estable sino que varios modos se excitan y se van desplazando ([18]).

Así pues hemos podido comprobar que el programa en MATLAB reproduce fielmente los principales resultados de las preguntas formuladas, empezando por el problema lineal, pasando por el problema FPU original, mostrando la super recurrencia, procesando los solitones *kink* de Toda, comprobando que el período de recurrencia se comporta como una potencia $N^{5/2}$ y proporcionando los resultados con una condición de inicio de Zabusky-Deem.

Parte III
Conclusión

CAPÍTULO 1

Conclusión

Para concluir este Proyecto de Fin de Máster vamos a repasar la metodología seguida y los resultados obtenidos. Como hemos visto, el problema de Fermi-Pasta-Ulam-Tsingou constituyó la primera simulación numérica de la historia de la Física. Se trata por lo tanto de un hito fundamental que no sólo demostró la utilidad de las simulaciones numéricas como una nueva y potente herramienta al servicio de la ciencia sino que además fue el punto de partida de varias teorías que con el tiempo se convirtieron en nuevas ramas de investigación de la Física, como los solitones o la teoría de Kolmogorov-Arnold-Moser (KAM).

Para la modificación del programa en MATLAB primero se ha analizado el programa básico partiendo de la formulación del problema FPU para así poder modificar el código y facilitar su ejecución, así como la comprensión de las diferentes problemáticas, como los solitones, la recurrencia y la super recurrencia, la amplitud creciente, la red de Toda o la excitación del modo superior. Así pues con este programa se puede estudiar el programa original y los desarrollos posteriores que constituyeron los inicios del análisis y la comprensión del problema FPU.

El hecho más evidente para demostrar la importancia del problema FPU, no ya en sus inicios como punto de partida de la física

computacional sino hoy en día, es que a pesar de haber transcurrido más de medio siglo desde entonces siguen escribiéndose artículos al respecto, no sólo recopilatorios en forma de artículos ([44], [45], [46], [47], [48], [49], [4], [3] y [50]) o de libros ([18] y [51]), sino también nuevas investigaciones sobre el problema y sobre su impacto en otras disciplinas ([52], [53], [54], [55], [56], [57], [58], [59], [60] y [61]).

Para entender la importancia de estos resultados basta con ver como los resultados de Fermi, Pasta y Ulam reavivaron el interés por la ecuación de Korteweg-de Vries, conocida como KdV. Como puede leerse en [50] la ecuación KdV es una ecuación diferencial no lineal que fue descubierta en 1895 por el físico holandés Diederick Korteweg y su estudiante Gustav de Vries para explicar los resultados presentados en 1844 por el ingeniero británico John Scott Russell sobre ondas en el canal Union en Escocia. Korteweg y de Vries encontraron una solución con una sola onda que concordaba con los experimentos de Russell, y también encontraron una solución periódica pero no fueron capaces de encontrar soluciones generales. Esta ecuación presentada en 1895 quedó en el olvido hasta que en los años 60 Zabusky primero ([6]) y después en colaboración con Kruskal ([7]) partieron del problema FPU pero con partículas y muelles infinitesimalmente pequeños para simular la deformación de un medio continuo. De este modo volvieron a derivar la ecuación KdV y con simulaciones numéricas generadas con la colaboración de Gary Deem ([14]) consiguieron observar la casi recurrencia y, para describir sus soluciones, acuñaron el término solitón que representaba la onda solitaria. Además mostraron que si la condición inicial era un modo con una longitud de onda corta el sistema sí que se comportaba de forma caótica mezclando energía entre los diferentes modos.

Además, la relación entre el problema FPU y la teoría KAM también es un punto clave ([51]). La teoría KAM mostró que a pesar de que las ecuaciones de movimiento no sean integrables para 3 o más partículas las perturbaciones pequeñas no causaban que el sistema se alejase del problema lineal. Afortunadamente otro resultado de Arnold demostró que a medida que el número de partículas aumenta el volumen

del espacio de fase que es estocástico aumenta, lo que se conoció como la difusión de Arnold. Pero un resultado posterior de Nekhoroshev mostró que la difusión era exponencialmente lenta en función del parámetro de perturbación, pero que podía hacerse grande si el número de grados de libertad aumentaba. Así pues esto mostraba que la equipartición de un sistema con N grados de libertad y una perturbación fija se alcanzaría a medida que N aumentase pero que el tiempo para alcanzarla aumentaba exponencialmente.

Como puede verse el problema FPU se presentaba como un problema perfecto para realizar simulaciones numéricas para comparar los resultados computacionales con las teorías de perturbación del hamiltoniano. La respuesta al problema dividió a los físicos en dos grupos contrapuestos: los que pensaban que era simplemente una curiosidad matemática que no se asemejaba a una realidad física y otro grupo propugnaba que se trataba de un problema fundamental entre la física clásica y la física estadística y cuántica. Pasados más de 50 años desde su descubrimiento todavía no se ha llegado a un consenso sobre una explicación físico-matemática teórica que explique el comportamiento observado en el problema FPU.

Aparte de estas áreas iniciales que surgieron a partir del problema FPU o que se desarrollaron ampliamente al estudiarlo como la Física Computacional, los solitones, la ecuación KdV y la teoría KAM, otras áreas también han surgido o se han desarrollado posteriormente gracias en parte al problema FPU. Las áreas principales son:

- la teoría del caos y especialmente la teoría del caos débil en los años 60 y posteriores. El problema FPU es un ejemplo claro de como una mínima variación en las condiciones iniciales afecta de forma mayúscula al comportamiento del sistema a largo plazo. En el caso del problema FPU se encontró que existía un umbral por debajo del cual el sistema tiene un comportamiento normal salvo por períodos breves caóticos, lo que se conoce como caos débil, y por encima de ese umbral aparece el caos fuerte lo que hace que la energía se reparta rápidamente entre las partículas ([57]),

- la teoría del metaequilibrio en los años 70 y posteriores, es decir que el problema FPU no alcanza un estado de equilibrio sino que no es más que un equilibrio ficticio o metaequilibrio y que para tiempos de simulación muy largos se llegaría a un equilibrio real predicho por la mecánica estadística clásica, es decir la *termalización* del sistema ([49]),
- la teoría de los modos localizados intrínsecos o *ILMs* en inglés por *intrinsic localized modes* también conocidos como *discrete breathers*, en los años 80 y posteriores. Se trata del equivalente en redes discretas extensas de los solitones que aparecen en sistemas continuos, pero a diferencia de los solitones no se propagan sino que vibran sin propagarse. Este tipo de ondas ya se han observado experimentalmente y se ha demostrado que pueden aparecer ILMs en la red del problema FPU ([53]),
- la teoría de los *q-Breathers* o QB en los años 2000, que se corresponden con oscilaciones periódicas de baja frecuencia que solventan el problema FPU no lineal y que están localizadas exponencialmente en el espacio de los modos normales q y son estables para valores pequeños del término no lineal, coincidiendo el umbral de estabilidad de las soluciones QB con el umbral del caos débil mencionado anteriormente ([56]).

Como puede verse el problema FPU fue el punto de partida de múltiples desarrollos posteriores tanto en la Física como en las Matemáticas y 60 años después sigue despertando la fascinación de los científicos que intentan todavía explicar de forma teórica el comportamiento del sistema que se estudió de forma numérica. La bibliografía listada en esta memoria y que puede encontrarse en el *USB* incluye los principales artículos para reconstruir la evolución del análisis de este problema y para ver los últimos desarrollos.

Parte IV

Bibliografía

Bibliografía

- [1] E. Fermi, J. Pasta, and S. Ulam. Studies of non linear problems. *Collected Papers of Enrico Fermi*, 2:978–988, 1965. Publicado originalmente en 1955 en *Los Alamos Report LA-1940*.
- [2] S. Ulam. Introduction no. 266. *Collected Papers of Enrico Fermi*, 2:977–978, 1965.
- [3] Thierry Dauxois. Fermi, Pasta, Ulam, and a mysterious lady. *Physics Today*, 61(1):55–57, 2008.
- [4] Thierry Dauxois, Michel Peyrard, and Stefano Ruffo. The Fermi-Pasta-Ulam numerical experiment: history and pedagogical perspectives. *European Journal of Physics*, 26(5):S3–S11, 2005.
- [5] J. L. Tuck and M. T. Menzel. The superperiod of the nonlinear weighted string (FPU) problem. *Advances in Mathematics*, 9(3):399–407, 1972.
- [6] Norman J. Zabusky. Exact solution for the vibrations of a nonlinear continuous model string. *Journal of Mathematical Physics*, 3(5):1028–1039, 1962.
- [7] Norman J. Zabusky and Martin D. Kruskal. Interaction of solitons in a collisionless plasma and the recurrence of initial states. *Physical Review Letters*, 15:240–243, 1965.
- [8] Nobuhiko Saitô and Hajime Hirooka. Long-time behavior of the vibration in one-dimensional harmonic lattice. *Journal of the Physical Society of Japan*, 23(2):157–166, 1967.

- [9] Nobuhiko Saitô and Hajime Hirooka. Computer studies of ergodicity in coupled oscillators with anharmonic interaction. *Journal of the Physical Society of Japan*, 23(2):167–171, 1967.
- [10] Joseph Ford. Equipartition of energy for nonlinear systems. *Journal of Mathematical Physics*, 2(3):387–393, 1961.
- [11] Morikazu Toda. Vibration of a chain with nonlinear interaction. *Journal of the Physical Society of Japan*, 22(2):431–436, 1967.
- [12] F. M. Izrailev and B. V. Chirikov. Statistical properties of a nonlinear string. *Doklady Akademii Nauk SSSR*, 166(1):57–59, 1966.
- [13] Morikazu Toda. *Theory of Nonlinear Lattices*, volume 20 of *Springer Series in Solid-State Sciences*. Springer, 1989. ISBN 978-3-642-83219-2.
- [14] Norman J. Zabusky and Gary S. Deem. Dynamics of nonlinear lattices I. Localized optical excitations, acoustic radiation, and strong nonlinear behavior. *Journal of Computational Physics*, 2(2):126–153, 1967.
- [15] David Sholl. Modal coupling in one-dimensional anharmonic lattices. *Physics Letters*, 149(5,6):253–257, 1990.
- [16] E. Atlee Jackson. Nonlinear coupled oscillators II. Comparison of theory with computer simulations. *Journal of Mathematical Physics*, 4(5):686–700, 1963.
- [17] R. L. Bivins, N. Metropolis, and J. Pasta. Nonlinear coupled oscillators: Modal equation approach. *Journal of Computational Physics*, 12:65–87, 1973.
- [18] Thomas P. Weissert. *The Genesis of Simulation in Dynamics - Pursuing the Fermi-Pasta-Ulam Problem*. Springer, 1997. ISBN 978-1-4612-1956-9.
- [19] Norman J. Zabusky. Phenomena associated with the oscillations of a nonlinear model string: The problem of Fermi, Pasta and Ulam. *Mathematical Models in Physical Science, Conference at the University of Notre Dame*, pages 99–133, 1962.

- [20] Norman J. Zabusky. Solitons and energy transport in nonlinear lattices. *Computer Physics Communications*, 5:1–10, 1973.
- [21] Martin D. Kruskal and Norman J. Zabusky. Stroboscopic perturbation procedure for treating a class of nonlinear wave equations. *Journal of Mathematical Physics*, 5:231–244, 1964.
- [22] F. D. Tappert and Norman J. Zabusky. Gradient-induced fission of solitons. *Physical Review Letters*, 27(26):1774–1776, 1971.
- [23] E. Atlee Jackson. Nonlinear coupled oscillators I. Perturbation theory; ergodic problem. *Journal of Mathematical Physics*, 4(4):551–558, 1963.
- [24] Joseph Ford and John Waters. Computer studies of energy sharing and ergodicity for nonlinear oscillator systems. *Journal of Mathematical Physics*, 4(10):1293–1306, 1963.
- [25] John Waters and Joseph Ford. A method of solution for resonant nonlinear coupled oscillator systems. *Journal of Mathematical Physics*, 7(2):399–403, 1966.
- [26] Grayson H. Walker and Joseph Ford. Amplitude instability and ergodic behavior for conservative nonlinear oscillator systems. *Physical Review*, 188(1):416–432, 1969.
- [27] Joseph Ford and Gary H. Lunsford. Stochastic behavior of resonant nearly linear oscillator systems in the limit of zero nonlinear coupling. *Physical Review A*, 1(1):59–70, 1970.
- [28] Gary H. Lunsford and Joseph Ford. On the stability of periodic orbits for nonlinear oscillator systems in regions exhibiting stochastic behavior. *Journal of Mathematical Physics*, 13(5):700–705, 1972.
- [29] A. N. Kolmogorov. Preservation of conditionally periodic movements with small change in the Hamilton function. *De Los Alamos Scientific Laboratory Translation LA-TR-71-67 de Helen Dahlby a partir del original Akad. Nauk. S.S.S.R. Doklady*, 98:527–532, 1954.
- [30] Jürgen Moser. On invariant curves of area-preserving mappings of an annulus. *Nachrichten der Akademie der Wissenschaften in Göttingen, Mathematisch-Physikalische*, 1:1–20, 1962.

- [31] Jürgen Moser. Remark on the paper “On invariant curves of area-preserving mappings of an annulus”. *Regular and Chaotic Dynamics*, 6(3):337–338, 2001.
- [32] V. I. Arnold. Proof of a theorem of Kolmogorov on the invariance of quasi-periodic motions under small perturbations. *Russian Math. Surveys*, 18(5):9–36, 1963.
- [33] V. I. Arnold. Small denominators and problems of stability of motion in classical and celestial mechanics. *Russian Math. Surveys*, 18(6):85–191, 1963.
- [34] V. I. Arnold. Instability of dynamical systems with several degrees of freedom. *Translation into English in Soviet. Math. Dokl. 5 (1964) del original en Dokl. Akad. Nauk. S.S.S.R.*, 156(1):9–12, 1964.
- [35] N. N. Nekhoroshev. Behavior of hamiltonian systems close to integrable. *Translated from Funktsionalnyi Analiz i Ego Prilozheniya*, 5(4):82–83, 1971.
- [36] Jürgen Pöschel. Nekhoroshev estimates for quasi-convex hamiltonian systems. *Math. Z.*, 213:187–216, 1993.
- [37] Henri Poincaré. Sur le problème des trois corps et les équations de la dynamique. *Acta Mathematica*, 13:1–270, 1890.
- [38] A. N. Kolmogorov. General theory of dynamic systems and classical mechanics. *Proceedings of the International Congress of Mathematicians, Amsterdam*, 1:315–333, 1957.
- [39] Michel Hénon and Carl Heiles. The applicability of the third integral of motion: some numerical experiments. *The Astronomical Journal*, 69(1):73–79, 1964.
- [40] F. G. Gustavson. On constructing formal integrals of a hamiltonian system near an equilibrium point. *The Astronomical Journal*, 71(8):670–686, 1992.
- [41] John M. Greene. Two-dimensional measure-preserving mappings. *Journal of Mathematical Physics*, 9(5):760–768, 1968.

- [42] Joseph Ford, Spotswood D. Stoddard, and Jack S. Turner. On the integrability of the Toda lattice. *Progress of Theoretical Physics*, 50(5):1547–1560, 1973.
- [43] Michel Hénon. Integrals of the Toda lattice. *Physical Review B*, 9(4):1921–1923, 1974.
- [44] Joseph Ford. The Fermi-Pasta-Ulam problem: Paradox turns discovery. *Physics Reports*, 213(5):271–310, 1992.
- [45] Richard S. Palais. The symmetries of solitons. *Bulletin (New Series) of the American Mathematical Society*, 34(4):339–403, 1997.
- [46] Norman J. Zabusky. Fermi-Pasta-Ulam, solitons and the fabric of nonlinear and computational science: History, synergetics, and visio-metrics. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 15:015102, 2005.
- [47] G. M. Zaslavsky. Long way from the FPU-problem to chaos. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 15(1):015103, 2005.
- [48] David K. Campbell, Phillip Rosenau, and George M. Zaslavsky. Introduction: The Fermi-Pasta-Ulam problem - the first fifty years. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 15(1):015101, 2005.
- [49] A. Carati, L. Galgani, and A. Giorgilli. The Fermi-Pasta-Ulam problem as a challenge for the foundations of Physics. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 15(1):015105, 2005.
- [50] Mason A. Porter, Norman J. Zabusky, Bambi Hu, and David K. Campbell. Fermi, Pasta, Ulam and the birth of experimental Mathematics. *American Scientist*, 97(3):214–221, 2009.
- [51] Giovanni Gallavotti. *The Fermi-Pasta-Ulam Problem*, volume 728 of *Lecture Notes in Physics*. Springer, 2008. ISBN 978-3-540-72995-2.
- [52] Lapo Casetti, Monica Cerruti-Sola, Marco Pettini, and E. G. D. Cohen. The Fermi-Pasta-Ulam problem revisited: Stochasticity thresholds in nonlinear hamiltonian systems. *Physical Review E*, 55(6):6566–6574, 1997.

- [53] David K. Campbell, S. Flach, and Yuri S. Kivshar. Localizing energy through nonlinearity and discreteness. *Physics Today*, 57(1):43–49, 2004.
- [54] Gianni Arioli, Hans Koch, and Susanna Terracini. Two novel methods and multi-mode periodic solutions for the Fermi-Pasta-Ulam model. *Communications in Mathematical Physics*, 255(1):1–19, 2005.
- [55] G. P. Berman and F. M. Izrailev. The Fermi-Pasta-Ulam problem: Fifty years of progress. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 15(1):015104, 2005.
- [56] S. Flach, M. V. Ivanchenko, and O. I. Kanakov. q-Breathers and the Fermi-Pasta-Ulam problem. *Physical Review Letters*, 95(6):064102, 2005.
- [57] Marco Pettini, Lapo Casetti, Monica Cerruti-Sola, Roberto Franzosi, and E. G. D. Cohen. Weak and strong chaos in Fermi-Pasta-Ulam models and beyond. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 15(1):015106, 2005.
- [58] A. Ponno and D. Bambusi. Korteweg-de Vries equation and energy sharing in Fermi-Pasta-Ulam. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 15(1):015107, 2005.
- [59] Ivan C. Christov. Hidden solitons in the Zabusky-Kruskal experiment: Analysis using the periodic, inverse scattering transform. *Mathematics and Computers in Simulation*, 82(6):1069–1078, 2010.
- [60] Miguel Onorato, Lara Vozella, Davide Proment, and Yuri V. Lvov. Route to thermalization in the α -Fermi-Pasta-Ulam system. *Proceedings of the National Academy of Sciences*, 112(14):4208–4213, 2015.
- [61] C. Danieli, D. K. Campbell, and S. Flach. Intermittent many-body dynamics at equilibrium. *Chaotic Dynamics*, pages 1–11, 2017.

Parte V
Apéndices

APÉNDICE A

Programas

En este apéndice incluiremos únicamente el código del programa. Aquí se podrá ver el detalle del código en Matlab incluyendo las modificaciones que se han realizado para su compilación y ejecución teniendo en cuenta qué aspectos querían simularse. Aunque este detalle pueda parecer innecesario es la mejor manera de asegurarse de que el código se conserva en formato impreso con los detalles utilizados, por si surgiesen otras versiones de éste o si se corrompiesen los archivos. Se muestra tanto el programa principal, `fpu.m`, como la función `fpu1.m` que se creó para ser llamada desde el código principal. Como puede comprobarse el código original ha sido modificado ampliamente para conseguir más versatilidad y para lograr un mejor manejo y unos resultados gráficos más fácilmente interpretables.

```
%-----%
% Main code for carrying out the numerical experiment
% Based on the code provided by Thierry Dauxois in
% http://perso.ens-lyon.fr/thierry.dauxois/fpu.html
%-----%

% Clear workspace
clear

% Activation of variables based on the question
q = 6;

% Initial variables from original FPU problem
N=32;           % Number of particles must be a power of 2
```

```
alpha=0.25;          % Non linear parameter
TMAX=10800;         % Lattice time for FPU recurrence
amax=1;            % Amplitude
DT=20;             % Time Step (delta t)

% Variables for the different scenarios to be test
% q = 0 - we remove the non linear term
% q = 1 - original FPU simulation
% q = 2 - superrecurrence
% q = 3 - increasing amplitude to observe energy equipartition
% q = 4 - Toda lattice
% q = 5 - Recurrence period follows  $N^{(5/2)}$ 
% q = 6 - Zabusky-Deem initial condition

if q == 0,
    alpha = 0;      % Linear systems
elseif q == 1,
    % Use original values listed above from FPU
elseif q == 2,
    N = 16;
    TMAX = 40000; % Lattice time for superrecurrence
    DT = 10;
elseif q == 3,
    amax = 10;     % increasing amplitude
elseif q == 4,
    N = 128;
    alpha = 0.25;
    TMAX = 400;
    DT = 5;
elseif q == 5,
% Different values depending on N
% We will run the program multiple times to record recurrence for each N
    N = 16;
    if N == 8,
        TMAX = 500;
        DT = 5;
        % Recurrence pick at 24 based on graphical value
    elseif N == 16,
        TMAX = 2000;
        DT = 10;
        % Recurrence pick at 60 based on graphical value
    elseif N == 32,
        % Keep original values
        % Recurrence pick at 158 based on graphical value
    elseif N == 64,
        TMAX = 60000;
        DT = 40;
        % Recurrence pick at 434 based on graphical value
    elseif N == 128,
        TMAX = 330000;
        DT = 80;
        % Recurrence pick at 1210 based on graphical value
    elseif N == 256,
        TMAX = 1880000;
        DT = 160;
        % Recurrence pick at 3394 based on graphical value
    end
elseif q == 6,
```

```

    N = 64;
    alpha = 0.25;
    TMAX = 64;
    DT = 2;
end

% Create time vector
tspan=[0:DT:TMAX];

% Test different tolerances, changing Reltol
if q == 0 | q == 1 | q == 2 | q == 3 | q == 5,
    for I=1:N,
        % Mode frequencies
        omegak2(I)=4*(sin(pi*I/2/N))^2;
        % FPU initial conditions
        % Increasing amplitude to show how the recurrence disappears
        for AT=1:amax,
            a=AT;
            b(I,AT)=a*sin(pi*I/(N+1));
            b(I+N,AT)=0;
        end
    end
end

if q == 4,
    for I=1:N,
        % Mode frequencies
        omegak2(I)=4*(sin(pi*I/2/N))^2;
        % Solitons
        k=0.4; sk=(sinh(k))^2; ek=exp(-k); i1=I-N/4; i2=i1-N/2;
        b(I,1)=-0.5/alpha*log((1+exp(2*k*(i1-1)))/(1+exp(2*k*i1)));
        b(I,1)=b(I)+0.5/alpha*log((1+exp(2*k*(i2-1)))/(1+exp(2*k*i2)));
        b(I+N,1)= sk*ek/alpha/cosh(k*i1)/(exp(-k*i1)+exp(k*i1)*exp(-2*k));
        b(I+N,1)=b(I+N,1)-sk*ek/alpha/cosh(k*i2)/(exp(-k*i2)+exp(k*i2)*exp(-2*k));
    end
end

if q == 6,
    for I=1:N,
        % Mode frequencies
        omegak2(I)=4*(sin(pi*I/2/N))^2;
        % Zabusky-Deem initial condition
        a=1;
        b(I,1)=a*sin(pi*N*I/(N+1));
        b(I+N,1)=0;
    end
end

% Setting options for ODE solving
options=odeset('Reltol',1e-4,'OutputFcn','odeplot','OutputSel',[1,2,N]);

% Close all previously created figures
close all;
% Solve Ordinary Differential Equation (ODE45 function in Matlab)
for AT=1:amax,
    [T,Y]=ode45(@(t,y) fpu1(t,y,N,alpha),tspan,b(:,AT)',options); % Time integration
    for IT=1:(TMAX/DT),
        TIME(IT)=IT*DT*sqrt(omegak2(1))/2/pi; % Time interation loop
    end
end

```

```

        YX(IT,1:N+1)=[0 Y(IT,1:N)]; YV(IT,1:N+1)=[0 Y(IT,N+1:2*N)];
        sXF(IT,:)=imag(fft([YX(IT,1:N+1) 0 -YX(IT,N+1:-1:2)]))/sqrt(2*(N+1));
        sVF(IT,:)=imag(fft([YV(IT,1:N+1) 0 -YV(IT,N+1:-1:2)]))/sqrt(2*(N+1));
        Energ(IT,1:N)=(omegak2(1:N).*(sXF(IT,2:N+1).^2+sVF(IT,2:N+1).^2)/2;
        for J=2:N-1,
            DifY(IT,J)=Y(IT,J+1)-Y(IT,J);
        end
    end
    if q == 3,
        figure;
        plot(TIME,Energ(:,1),TIME,Energ(:,2),TIME,Energ(:,3),TIME,Energ(:,4),TIME,Energ(:,5));
        figure;
        surf(DifY); % Space derivative field to show the soliton dynamics
    end
end

if q == 0 | q == 1 | q == 2 | q == 4 | q == 5,
    % For the linear problem we only plot the space derivative
    if q ~= 0,
        plot(TIME,Energ(:,1),TIME,Energ(:,2),TIME,Energ(:,3),TIME,Energ(:,4),TIME,Energ(:,5));
    end
    % For the recurrence period we do not plot the space derivative
    if q ~= 5,
        figure;
        surf(DifY); % Space derivative field to show the soliton dynamics
    end
end

if q == 6,
    % We plot the higher modes based on the initial Zabusky-Deem condition
    plot(TIME,Energ(:,N),TIME,Energ(:,N-1),TIME,Energ(:,N-2),TIME,Energ(:,N-3),TIME,Energ(:,N-4),TIME,Energ(:,N-5));
    figure;
    surf(DifY); % Space derivative field to show the soliton dynamics
end

if q == 1,
    % Get crest of the waves removing the picks at the boundaries
    dini = max(max(DifY(:,3:29)));
    count = 0;
    % Select the movement of the wave by looking for the moving crest
    for IT=1:(TMAX/DT),
        if abs(max(DifY(IT,3:29))-dini)/dini < 0.15
            count = count+1;
            Difini(count,:)=DifY(IT,:);
            Xaxe(count)=IT;
        end
    end

    figure;
    % Plot the wave in time
    Yaxe=[1:31];
    surf(Yaxe,Xaxe,Difini);

    figure;
    %Plot the wave in numbers
    surf(Difini);

    %Create two separate movies of the opposing waves

```

```
IT1=[1,3,4,6,8,10,12,15,16,18,20,21,23];
IT2=[2,5,7,9,11,13,14,17,19,22];

figure;
% Number of frames per plot for graph
FR = 6;
% Create the graphs for the first wave
for J=1:length(IT1),
    plot(Difini(IT1(J),:));
    % In order to plot in slow motion each plot will appear in several frames
    for I=1:FR,
        M1(FR*(J-1)+I)=getframe(gcf);
    end
end

% Create the movie for the first wave
movie2avi(M1,'SolitonM1.avi');

figure;
% Create the graphs for the second wave
for J=1:length(IT2),
    plot(Difini(IT2(J),:));
    % In order to plot in slow motion each plot will appear in 24 frames
    for I=1:FR,
        M2(FR*(J-1)+I)=getframe(gcf);
    end
end

% Create the movie for the second wave
movie2avi(M2,'SolitonM2.avi');

figure;

% Create combined counters with same length for waves
ITC1=[1,3,4,6,8,10,12,15,16,18,20,21,23];
ITC2=[2,5,5,7,9,11,13,14,14,17,19,19,22];
% Create the graphs for the combined waves
for J=1:length(ITC1),
    plot(Difini(ITC1(J),:)+Difini(ITC2(J),:));
    % To plot in slow motion each plot will appear in multiple frames
    for I=1:FR,
        MC(FR*(J-1)+I)=getframe(gcf);
    end
end

% Create the movie for the combined waves
movie2avi(MC,'SolitonMC.avi');

% Create animated gif for the first wave
figure;
axis tight
set(gca,'nextplot','replacechildren','visible','off')

[im1,map1] = rgb2ind(M1(1).cdata,512,'nodither');

for K = 1:length(IT1)
    im1(:,:,1,K) = rgb2ind(M1(K*FR).cdata,map1,'nodither');
    pause(1);
end
```

```

end

imwrite(im1,map1,'SolitonM1.gif');

% Create animated gif for the second wave
figure;
axis tight
set(gca,'nextplot','replacechildren','visible','off')

[im2,map2] = rgb2ind(M2(1).cdata,512,'nodither');

for K = 1:length(IT2)
    im2(:,:,1,K) = rgb2ind(M2(K*FR).cdata,map2,'nodither');
    pause(1);
end

imwrite(im2,map2,'SolitonM2.gif');

% Combined animated gif for both waves
figure;
axis tight
set(gca,'nextplot','replacechildren','visible','off')

[imc,mapc] = rgb2ind(MC(1).cdata,512,'nodither');

for K = 1:length(ITC1)
    imc(:,:,1,K) = rgb2ind(MC(K*FR).cdata,mapc,'nodither');
    pause(1);
end

imwrite(imc,mapc,'SolitonMC.gif');

end

%-----%
% fpu1 function
%-----%
function dy=fpu1(t,y,N,alpha)

% Variables N and alpha are passed from the main function
% instead of redeclaring
% N=256;    % Number of particles must be a power of 2
% alpha=0.5; % Non linear parameter

D(N+1)=y(2)-2*y(1)+alpha*((y(2)-y(1))^2-y(1)^2);D(1)=y(N+1);
D(2*N)=y(N-1)-2*y(N)+alpha*(y(N)^2-(y(N)-y(N-1))^2);D(N)=y(2*N);

for I=2:N-1,
    D(N+I)=y(I+1)+y(I-1)-2*y(I)+alpha*((y(I+1)-y(I))^2-(y(I)-y(I-1))^2);
    D(I)=y(N+I);
end
dy=D';

```

APÉNDICE B

Archivos

En este apéndice incluiremos la estructura de los archivos que se han creado para este proyecto, en especial el programa principal en MATLAB (`fpu.m`), las funciones en MATLAB (`fpu1.m`), las condiciones iniciales en MATLAB (`fpu_initial.m`), el resumen de las opciones en MATLAB (`fpu_summary.m`), los resultados numéricos y gráficos (archivos de imagen `.png` y de vídeo `.mp4`) y finalmente el archivo de la memoria `principal.pdf` y otros archivos en formato `.pdf` como el diagrama de Gant (`gantts2.pdf`), el ordenador portátil (`hp.pdf`) y el logotipo de la Universidad (`uned.pdf`). Todos estos archivos se incluirán en un *USB* que acompañará la memoria.

Lo que no se incluirá es el detalle de los archivos y carpetas en \LaTeX para generar la memoria.

La nomenclatura de los archivos de imagen y de vídeo está construida para poder determinar a qué ejecución corresponden. Por ejemplo el archivo `figureXX_*.png` o `figureXX_*.mp4` se corresponde con el capítulo `XX`, o puede corresponderse con la sección como por ejemplo el apéndice 2 que se escribirá `ap02`, y el sufijo `*` será el número del archivo, por ejemplo `01`, o también la opción de ejecución con `q=1` correspondiendo a `q1`, o el número de partículas de una opción determinada como por

ejemplo `q5N8`, o la amplitud de una opción como por ejemplo `q3a01`. También puede usarse el sufijo para indicar otras opciones como `q5log` para la gráfica logarítmica de la opción `q=5`, o si es el problema original `q1fpu`, el de diferencias `q1dif`, el de solitones `q1sol` o un vídeo `q1vid`.

Otra cosa a mencionar es que estos archivos se adjuntarán a este pdf. Esto no se puede hacer en \LaTeX por lo que una vez se haya generado el pdf se adjuntará un archivo zip en el pdf. Puesto que el adjuntar un archivo puede causar errores al intentar abrirlo desde el pdf, el método más sencillo es renombrar el archivo como txt, por lo que luego será necesario salvarlo desde el pdf, cambiar la extensión de txt a zip y descomprimirlo.

Así pues los archivos contenidos en el zip se pueden ver a continuación, y también se incluye este zip como txt adjunto al pdf.

En cuanto a la bibliografía también se han incluido copias de todos los artículos relevantes pero éstos no se incluyen en el zip sino que se guardan por separado, puesto que su tamaño combinado imposibilitaría el manejo del archivo de la memoria. Así pues se entregará un *USB* que contendrá esta documentación.

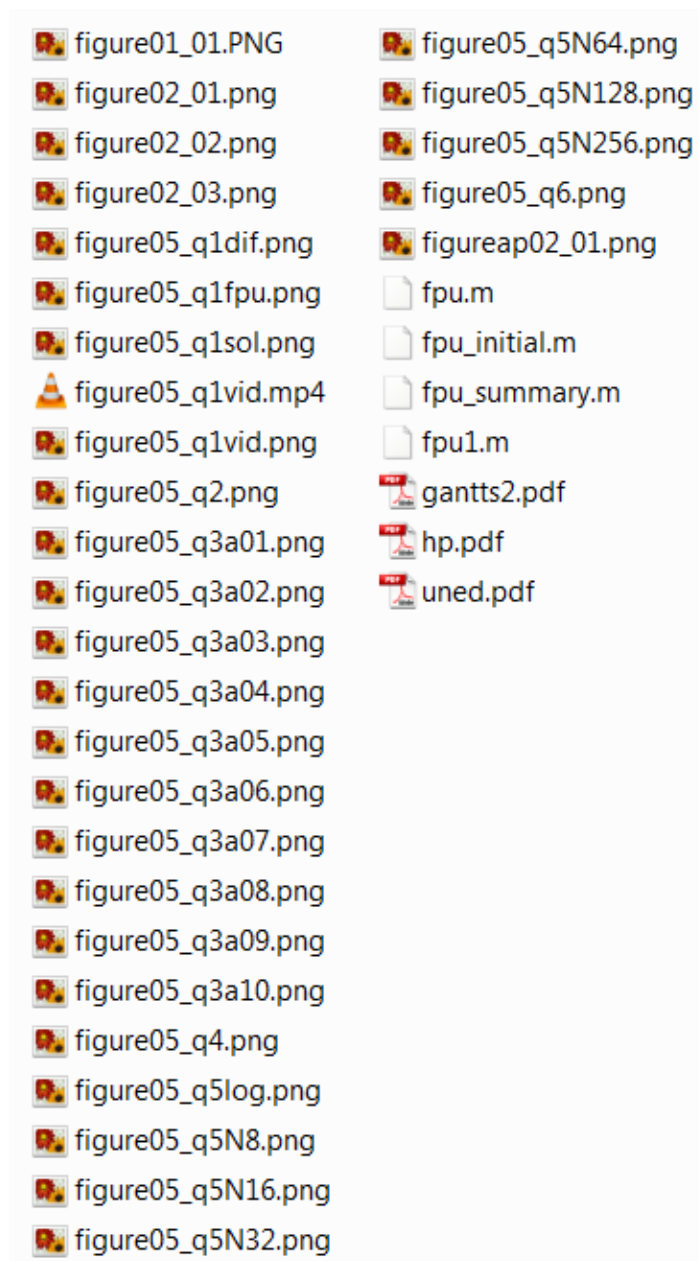


Figura B.1: Lista de archivos adjuntos

APÉNDICE C

Etapas de Desarrollo del Proyecto

En este apartado se incluirá una explicación de las diferentes etapas que condujeron a la conclusión de este proyecto. Primero se detallarán las etapas principales del proyecto que se ilustrarán posteriormente mediante un diagrama de Gantt explicativo que describirá la duración y las relaciones entre estas fases.

C.1 Etapas

C.1.1 Investigación bibliográfica

En todo proyecto la primera fase es siempre la búsqueda de información relevante. De esta manera, se estudian las bases sobre las que reposará todo el armazón futuro del proyecto, básicamente revisando los artículos y libros de los autores que han escrito sobre el tema. Además de en las bibliotecas y en las revistas científicas sobre el tema, es posible hoy en día acceder a casi toda la información a través de Internet. Se pueden consultar artículos publicados sobre esta materia e incluso libros enteros, encontrar sitios web divulgativos, y ejemplos que aclaren los conceptos fundamentales. Visto el componente informático y el uso de MATLAB como lenguaje de programación en este proyecto, se dispone de mucha información, de calidad, sobre este tema.

C.1.2 Estudio en profundidad

La cantidad y diversidad de información es tal que no sólo es importante buscar sino también saber escoger aquello que realmente podría ser útil. Una vez recopilados todos los datos necesarios se ha de repasar y estudiar cuidadosamente toda la información relevante para el proyecto. Empieza aquí una etapa de lectura y de previsión de qué derrotero tomará el proyecto en el futuro.

C.1.3 Perspectiva histórica

En este tipo de proyectos es importante repasar los primeros modelos físicos y los métodos matemáticos, algorítmicos y, por supuesto, informáticos para hacerse una idea de cuales han sido las aproximaciones y técnicas más usadas y como han evolucionado y evolucionan. Esto constituirá la primera parte de la memoria descriptiva.

C.1.4 Modelización del problema

A partir de la perspectiva histórica del programa se llega a los modelos utilizados para la resolución de este tipo de problemas. Principalmente se escogen y definen las variables principales y se escriben las ecuaciones maestras del problema, efectuando las aproximaciones y conjeturas necesarias para llegar a un modelo que pueda resolverse (analítica o, como en este caso, numéricamente) para comparar estos resultados con los valores documentados.

C.1.5 Análisis del programa existente

Puesto que ya se dispone de un programa en MATLAB que genera soluciones numéricas del problema de Fermi-Pasta-Ulam (FPU) analizaremos como está estructurado este programa con respecto a los modelos analizados anteriormente, qué aproximaciones utiliza y qué algoritmos se han implementado para resolver las ecuaciones.

C.1.6 Ejecución del programa

Conociendo la estructura y la algorítmica del programa lo ejecutaremos con diferentes valores de los parámetros. De esta manera recopilaremos juegos de datos de las ejecuciones efectuadas que nos servirán para su posterior análisis en el apartado siguiente.

C.1.7 Estudio de los resultados

A partir de los resultados numéricos obtenidos en la sección anterior, analizaremos cómo se comportan los diferentes métodos, los tiempos de ejecución y la concordancia con los resultados presentes en la literatura existente, mostrado también las gráficas de los resultados obtenidos.

C.1.8 Conclusión

A la conclusión de todas estas etapas ya está terminado el proyecto y pueden empezarse a elaborar los documentos definitivos.

C.2 Diagrama de Gantt

Este diagrama de Gantt detalla temporalmente las diferentes fases anteriormente descritas.

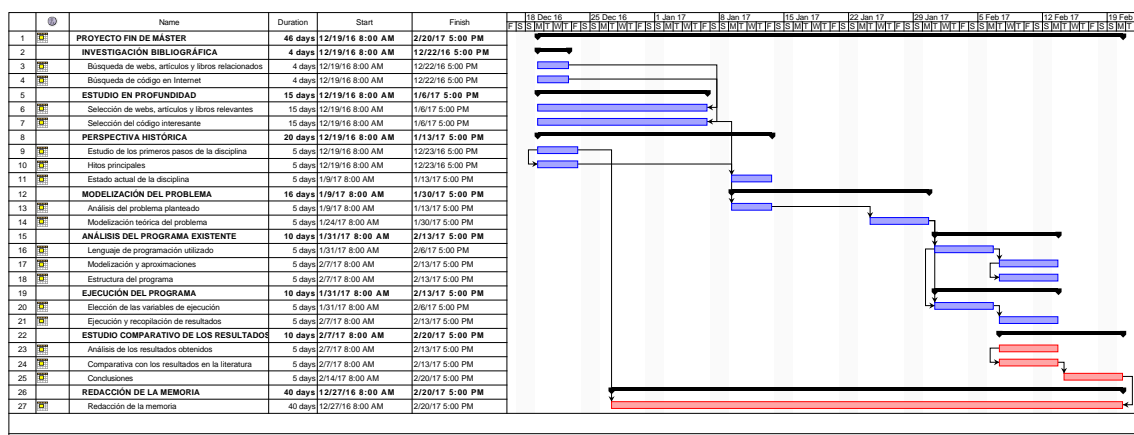


Figura C.1: Diagrama de Gantt del Proyecto Fin de Máster

APÉNDICE D

Presupuesto y Material

Para la elaboración de este proyecto, el material y el presupuesto necesarios pueden detallarse de forma muy sencilla. Al ser un proyecto enteramente teórico e informático, sólo hay tres factores que influyen: las personas involucradas, el material bibliográfico y el material informático necesario.

D.1 Material

El material utilizado en este proyecto, aparte algún soporte informático (*USB* y *CD*) y las cuartillas de papel, ha sido principalmente un ordenador personal portátil marca *HP* modelo *EliteBook 820*. Este ordenador fue adquirido hace 2 años, por un importe de 1.400€. Dada la rápida depreciación de los equipos informáticos su valor actual debe de rondar los 700€ (amortización en 4 años). Sus características son las siguientes:



Figura D.1: Ordenador portátil *HP EliteBook 820*

Características del <i>HP EliteBook 820</i>	
Procesador	Intel Core i5-5300U 2,30GHz
Disco Duro	256GB
RAM	8GB
Tarjeta Gráfica	8GB

Como se puede apreciar, tanto por sus características como en la foto, para la elaboración del proyecto no se necesita ningún equipo costoso, en especial comparado con la computadora original, la MANIAC I. Evidentemente, para su posterior puesta en marcha, cuanto más potente sea el equipo menor será el tiempo de ejecución de los programas.

D.2 Presupuesto

Ya se ha visto en el apartado anterior que el único material utilizado ha sido un ordenador con 2 años de antigüedad con un valor actual en torno a los 700€. Hay que añadir a esto el consumo eléctrico del equipo pero este gasto es despreciable comparado con el resto de costes.

Para estimar el coste de *mano de obra* se necesita saber cuál es el sueldo medio horario de una persona que realiza este tipo de tareas, como podría ser un becario o investigador. Si optamos por la opción más económica (la del becario), con un coste de 400€ al mes, puesto que el Proyecto de Fin de Máster consta de 12 créditos, es decir 300 horas lo cual equivale aproximadamente a 8 semanas de trabajo a jornada completa, esto nos daría un coste en personal de unos 800€.

Como se ve el coste es bajo, unos 1.500€ puesto que se ha encontrado información sobre la formulación básica del problema FPU, por lo que la mayor parte del tiempo se ha empleado buscando documentación, entendiendo la teoría, analizando y extendiendo el código, realizando pruebas y redactando esta Memoria.