



Modelamiento de ondas largas con pequeñas amplitudes unidireccionales

Antonio Vásquez, Joel Javier

Orientador: Prof Dr. Dugán Paúl Nina Ortiz

Tesis profesional presentada al Programa Profesional de Ciencia de la Computación como parte de los requisitos para obtener el Título Profesional de Lic. en Ciencia de la Computación.

UCSP- Universidad Católica San Pablo
Diciembre de 2017

A mis padres y hermanos, por su infinito apoyo y amor a lo largo de mi vida. A mi orientador, por las discusiones e ideas iluminadoras que compartimos.

UNIVERSIDAD CATÓLICA SAN PABLO
FACULTAD DE INGENIERÍA Y COMPUTACIÓN
PROGRAMA PROFESIONAL DE CIENCIA DE LA
COMPUTACIÓN

Modelamiento de ondas largas con pequeñas
amplitudes unidireccionales

Tesis de graduación presentado por el bachiller Antonio Vásquez,
Joel Javier en el cumplimiento de los requisitos para obtener el
título profesional de Licenciado en Ciencia de la Computación.

Arequipa, December 13, 2017

Aprobado por:

Prof. Dr. Hidalgo Buena Gente
PRESIDENTE

Prof. Dr. Manuel Armando Líos
SECRETARIO

Prof. Dr. Antero A. Gal Oppe
INTEGRANTE

Prof. Dr. Casso E. Staria
EXTERNO
Universidad del ABC

Abreviaturas

PCU *Propiedad de Continuación Única*

MEF *Método de Elementos Finitos*

KdV *Korteweg—de Vries*

BBM *Benjamin—Bona—Mahony*

Agradecimientos

A mis padres y hermanos, por su infinito apoyo y amor a lo largo de mi vida. A mi orientador, por las discusiones e ideas iluminadoras que compartimos.

Resumen

Dada la complejidad de comprobar la estabilización de la ecuación Benjamin–Bona–Mahony, L. Rosier propuso una ecuación que se aproxima a esta para encontrar soluciones en la ecuación original. Al no existir una solución puramente matemática en la ecuación BBM, se propone el análisis numérico basado en la técnica de Elementos Finitos y describir el comportamiento de ondas largas con pequeñas amplitudes unidireccionales en la ecuación propuesta por L. Rosier para comprobar la convergencia en un espacio rectangular.

Abstract

Since the complexity of the evidence in the stabilization of the Benjamin–Bona–Mahonys' equation, L. Rosier proposed an equation which aproximes to the first equation in order to find solutions in the original one. Since there is no a pura mathematical solution for the BBM equation, we propose a numerical analysis based on the Finite Element method and then describe the behavior of long waves with small unidirectional amplitudes in Rosier's equation in order to prove the convergence into a rectangular space.

Contents

1	Introducción	2
1.1	Motivación y Contexto	2
1.2	Planteamiento del Problema	3
1.3	Objetivos	3
1.3.1	Objetivo General	3
1.3.2	Objetivos Específicos	3
1.4	Organización de la tesis	3
2	Estado del arte	5
2.1	Estado del arte	5
2.1.1	Método de Elementos Finitos (<i>Método de Elementos Finitos</i> (MEF))	5
2.1.2	Métodos que se aproximan a MEF	6
3	Marco Teórico	8
4	Propuesta	10
4.1	Algoritmo Secuencial	10
4.2	Algoritmo Paralelizado	11
5	Pruebas y Resultados	13
5.1	Con pequeñas cantidades de datos, para $a = -5$ y $b = 15$	13
5.2	Con grandes cantidades de datos, para $a = -100$ y $b = 100$	16
5.3	Para 10 ondas simultáneas, con $a = -5$ y $b = 95$	17

5.4	Comprobación gráfica del océano real y numérico	18
5.5	Para 1000 ondas simultáneas, con $a = -100$ y $b = 100$	20
6	Conclusiones y Trabajos Futuros	22
6.1	Problemas encontrados	22
6.2	Recomendaciones	23
6.3	Trabajos futuros	23
	Bibliografía	28

List of Tables

5.1	Comparación entre la solución numérica y analítica con $c = 0.001, \tau = 0.00001, h = 0.1$ en $t = 0.01, 0.05, 0.1$	14
5.2	Comparación entre la solución numérica y analítica con $c = 0.001, \tau = 0.00001, h = 0.1$ en $t = 0.01, 0.05, 0.1$	16

List of Figures

5.1	Función analítica y numérica del ejemplo con $c = 0.001, \tau = 0.00001, h = 0.1$ en $t = 0.1$	14
5.2	Función analítica y numérica del ejemplo con $c = 0.001, \tau = 0.00001, h = 0.1$ en $t = 0.01$	15
5.3	Función analítica y numérica del ejemplo con $c = 0.001, \tau = 0.00001, h = 0.1$ en $t = 0.05$	15
5.4	Función analítica y numérica del ejemplo con $c = 0.001, \tau = 0.00001, h = 0.1$ en $t = 0.05$	16
5.5	Función numérica del ejemplo con $c = 0.001, \tau = 0.00001, h = 0.1$ en $t = 0.01$, para un total de 10 ondas	17
5.6	Función numérica del ejemplo con $c = 0.001, \tau = 0.00001, h = 0.1$ en $t = 0.1$, para un total de 10 ondas	18
5.7	O_N detenido en $t = 50$	19
5.8	Campo de vectores de O_R y O_N	20
5.9	Función numérica del ejemplo con $c = 0.001, \tau = 0.00001, h = 0.1$ en $t = 0.1$, para un total de 1000 ondas	21

Chapter 1

Introducción

1.1 Motivación y Contexto

En el modelamiento de propagación de ondas, es críticamente importante cómo las fronteras dinámicas influyen en la estabilización de tales propagaciones. Debido a la complejidad de la gobernación de la estabilización, muchos modelos han tratado de brindar predicciones, trabajando en diferente tipos de espacios configurados (e.g., Torus de una-dimensión $\mathbb{T} = \mathbb{R}/2\pi$) con el objetivo de obtener algunos resultados que puedan ayudar a la predicción de ondas largas con pequeñas amplitudes en medios de dispersión no lineales.

En 1894, Gustav de Vries publicó en su tesis [Vr94] un modelo llamado, la ecuación de *Korteweg—de Vries* (KdV),

$$u_t + u_{xxx} + u_x + uu_x = 0, \quad (1.1)$$

para describir la propagación de ondas unidireccionales largas con amplitudes pequeñas en medios de dispersión no lineales con una dimensión. Más adelante, Benjamin et al. [BBM72] propusieron un modelo llamado, la ecuación de *Benjamin—Bona—Mahony* (BBM), como una ecuación regularizadora de ondas largas,

$$u_t - u_{txx} + u_x + uu_x = 0, \quad (1.2)$$

donde el término de dispersión $-u_{txx}$ en la Ecuación (1.2) tiene un fuerte efecto suave para la regularidad del tiempo, que se convierte en una teoría bien fundamentada para la ecuación, y de forma más fácil de manejar que la Ecuación (1.1) como podemos ver en [BoTz09] y [Ro10].

En la actualidad, los matemáticos se están enfocando en probar la controlabilidad y estabilización de la ecuación BBM (ver [RoZh12], [Mi01] y [ZhZu03]). Más específicamente, están enfocados en probar la conjetura de la *Propiedad de Continuación Única* (PCU) enunciado por L. Rosier [Ro16].

Discretizar el algoritmo para computarlo y obtener un mejor modelado de la Ecuación (1.1), nos ayudará a una mejor simulación de ondas en tiempo real, como fue mostrado por

Namkyung et al. [LBR07]. Dado que aún no existe una solución puramente matemática para la estabilización de ondas largas con pequeñas amplitudes debido a la alta complejidad matemática en el manejo de los operadores de la ecuaciones, nos enfocamos en el Análisis Numérico de la Ecuación (1.1) para discretizar el algoritmo de forma numérica en el sentido de diferencias divididas, para obtener un mejor modelamiento de ondas largas. En efecto, trabajamos sobre un espacio $(\mathbb{R} \times \mathbb{R}_+)$ con una asintótica computacional $\mathcal{O}(h^2)$ donde h es $(b - a)$ para $b \in \mathbb{R}$ y $a \in \mathbb{R}_+$.

1.2 Planteamiento del Problema

Conjetura de la Propiedad de Continuación Única (PCU) para la ecuación BBM, enunciado por L. Rosier en el 2016 [Ro16], donde se pone a juicio la estabilización de la ecuación BBM. Por tal motivo, no se define un espacio común y general para el estudio de la Ecuación (4.1), por lo que no se tiene en cuenta las fronteras fijas para la estabilización.

Observación: El estudio numérico de la Ecuación (1.1) de la KdV, se ha dado desde muchos antes que L. Rosier enuncie la Conjetura PCU. Sin embargo, el análisis numérico de la Ecuación (4.1) aún no se había desarrollado. En efecto, el análisis más aproximado de dicha ecuación fue dada por M. Xiao et al. en el 2011 [Ro10]. Por lo que el autor se enfoca en la adaptación del Problema 1.2 para la Ecuación (4.1).

1.3 Objetivos

1.3.1 Objetivo General

Modelar ondas largas con pequeñas amplitudes unidireccionales.

1.3.2 Objetivos Específicos

- Construir el análisis numérico para la ecuación propuesta por Lionel Rosier [Ro16] para un espacio rectangular.
- Presentar un algoritmo paralelo para ejecutar múltiples ondas al mismo tiempo.
- Predecir ondas a través del tiempo en un espacio rectangular.

1.4 Organización de la tesis

Este trabajo está organizado de la siguiente forma. En el Capítulo 2, se presenta el estado del arte basado en técnicas que se aproximan al método de Elementos Finitos para la ecuación propuesta por L. Rosier, dado que no existe trabajos que han planteado

soluciones directamente a la ecuación, solo se describe la técnica e idea utilizada en los trabajos mencionados. Así mismo, en el Capítulo 3 se da a conocer el Marco Teórico basado en la teoría matemática que es la base del análisis, para luego pasar a describir la solución numérica del algoritmo y su convergencia en el Capítulo 4. De igual forma, se presenta un algoritmo paralelo para lanzar múltiples ondas de forma simultánea en la misma capítulo. Luego, en el Capítulo 5 se presentan resultados de la convergencia del algoritmo para la ecuación sobre un espacio rectangular y la simulación de un océano gracias a la modelación de múltiples ondas. Además, se muestran los resultados para el algoritmo en tiempo lineal y para el algoritmo paralelizado, para luego pasar a las conclusiones y trabajos futuros en el Capítulo 6. Por último, se presenta un Apéndice 6.3 con información adicional y Teoremas matemáticos utilizados a lo largo del trabajo.

Chapter 2

Estado del arte

2.1 Estado del arte

2.1.1 Método de Elementos Finitos (MEF)

Kappor et al. [KRD12] usaron el algoritmo B-Spline MEF para la solución de la ecuación KdV, donde mejoraron el modelamiento de ondas a través de la reducción de la decadencia del Espacio de Energía, cuyo objetivo general es resolver el problema anterior mediante la simulación numérica, en la que se examina la exactitud de las soluciones calculadas mediante la comparación con soluciones analíticas. De esta forma, se discutió la interpolación física del estudio de *solitones*, en donde se encuentra que las ondas de viaje alcanzan la máxima magnitud de la velocidad en un corto tiempo del intervalo y hay una incertidumbre del movimiento de las ondas, después de añadir una nueva variable que depende de la evolución en el tiempo y sirve como regulador de la convergencia de la ondas para que se puedan obtener soluciones numéricas aproximadas a las soluciones numéricas dentro del espacio.

Más adelante, Batall y Turgut [BaTu16] usaron algunos resultados previos en la solución del método de Galerkin para la solución de la ecuación de Roseneau [KiLe99], para mejorar específicamente la ecuación Rosenau-KdV, con el objetivo de que el modelamiento funcione de forma adecuada, no sólo para ondas largas, sino de igual manera para amplitudes pequeñas de estas mismas. Asimismo, se basó en un subdominio usando el método de B-spline de orden sexto, donde se utiliza para simular el movimiento de onda solitaria. Además, las dos invariantes del movimiento se elaboran para definir las propiedades de conservación. Las normas de error \mathcal{L}^2 y \mathcal{L}^∞ , se utilizan para medir las diferencias entre las soluciones analíticas y numéricas, para poder evaluar el error de convergencia entre espacios.

Ninguno de estos resultados fueron capaces de obtener una adecuada aproximación para el medio dispersivo, al contrario de Ying y Piao [YiPi13], donde obtuvieron una buena aproximación usando el algoritmo cuadrático B-Spline MEF para la ecuación BBM con el algoritmo MEF en tiempo de complejidad de $\mathcal{O}(h)$, donde se consideraron dos ejemplos

para demostrar la eficacia del esquema propuesto, lo que permitió que soluciones numéricas obtenidas para diferentes viscosidades se compararan con las soluciones exactas. Los resultados numéricos muestran que el esquema es eficiente y factible, pero sin ninguna estabilización en el espacio trabajado. Por último, Cerezo y Serón [CeSe02] fueron capaces de renderizar ondas naturales gracias a la unión de Computación Gráfica usando la Ecuación (1.1), con la finalidad de obtener un comportamiento espectral de la absorción (i.e., soluciones exactas de las alturas de las ondas) y los coeficientes de dispersión para el tratamiento correcto de la fase de función en el proceso de generación de imágenes. Sin embargo, estos renderizados no fueron capaces de predecir el comportamiento de ondas largas y solo lograron modelar una pequeña dispersión en las amplitudes de las ondas.

2.1.2 Métodos que se aproximan a MEF

Zabusky y Kruskal [ZaKr65] descubrieron que la ecuación KdV tiene una solución *soliton* (i.e., una ola solitaria de auto-refuerzo, que mantiene su forma mientras se propaga a una velocidad constante). Además, ellos obtuvieron soluciones numéricas en espacios periódicos de la ecuación KdV e investigaron la interacción de ondas solitarias. Así, los solitones se separan. Debido a la periodicidad, dos o más solitones eventualmente se superponen espacialmente e interactúan de forma no lineal. Poco después de la interacción, reaparecen virtualmente sin afectar en tamaño o forma. En otras palabras, los solitones *pasan* uno al otro sin perder su identidad. Gracias a ello, se tiene un proceso físico no lineal en el cual los pulsos localizados son interactivos y no se dispersan nunca.

Gardner et al. [GGKM67] determinaron la relación entre la ecuación KdV y la ecuación lineal de Sturm–Liouville y obtuvieron un algoritmo para la solución de la ecuación KdV. Más adelante, sus métodos fueron extendidos por Gardner [Ga74], donde se tomaron condiciones de conjuntos con bordes y se trató el problema de valor inicial para la ecuación KdV con datos iniciales $u(x, 0) = u_0(x)$, donde $u_0(x)$ se asume que es limitado y tres veces continuamente diferenciable. El método de Gardner et al. ha demostrado ser aplicable a otras ecuaciones no lineales de evolución en el tiempo. Particularmente, ha sido aplicable a ecuaciones no lineales de Schrödinger por Zakharov y Shabat [ZaSh72], donde se describe el auto-enfoque plano y la auto-modulación unidimensional se puede resolver de forma exacta, reduciendo el problema original al problema de dispersión inversa para un operador diferencial lineal determinado. En este caso, se pueden obtener soluciones exactas que describen la interacción de paquetes de ondas solitarias–solitones. Se muestra que la amplitud y velocidad de los solitones que interactúan no cambian, mientras que la fase tiene una discontinuidad. También se demuestra que solo se producen colisiones de parejas de solitones. Los resultados obtenidos se utilizan para evaluar la fase no lineal del desarrollo de la inestabilidad auto-modulada.

Basado en la idea de semigrupos de operadores de ecuaciones no lineales, presentado por Lax [La68] y Bhatnagar [Bh79], se obtuvieron avances en la solución de ecuaciones no lineales e unicidad por Ablowitz y Segur [Bh79], donde ciertos problemas no lineales tienen una estructura subyacente sorprendentemente simple, y pueden resolverse por métodos esencialmente lineales. Típicamente, estos problemas están en forma de ecuaciones de evolución (e.g., ecuación diferencial estocástica), que describen cómo una variable (o un conjunto de variables) evoluciona en el tiempo a partir de un estado inicial dado. Es-

tas ecuaciones pueden tomar una variedad de formas, incluyendo ecuaciones diferenciales parciales, formas diferenciales (e.g., espacio discreto, tiempo continuo), diferencia parcial (e.g., tiempo y espacio discretos), integro–diferencial, así como ecuaciones diferenciales ordinarias acopladas (de orden finito). Aunque estos problemas sean no lineales, se puede obtener la solución general que se desprende de datos iniciales arbitrarios (dentro de una clase apropiada) sin aproximación. De igual forma, estos problemas exactamente solubles surgen naturalmente como modelos de fenómenos físicos (e.g., ondas largas con pequeñas amplitudes unidireccionales).

Whitham [Wh74] expandió las técnicas sobre solitons para ecuaciones de tipo onda no lineal, basado en métodos de teoría de dispersión sobre espaciones compactos con bordes. Más adelante, Dood et al. [DEGM84] realizaron una actualización de estas técnicas basado en espacios de Sobolev. Gracias a Lamb [La80] y Drazin [Dr83], se obtuvieron otras técnicas para resolver una clase más general de ecuaciones no lineales de evolución en el tiempo, donde relacionaron ciertas ecuaciones diferenciales lineales conocidos como ecuaciones Sturm–Liouville, donde después de reducir el orden de las ecuaciones, pudieron hacer un mapeamiento de las soluciones lineales a las soluciones de las ecuaciones diferenciales no lineales. Todos estos trabajos trataron de aproximar el MEF para comparar la Ecuación (1.2), pero sin ningún resultado óptimo en la parte de la *estabilización* de la parte no lineal para ondas largas y pequeñas amplitudes.

Chapter 3

Marco Teórico

En el presente capítulo, utilizamos la idea de quasi-interpolaciones, con la siguiente definición.

Definición 1. *Dado un $f^*(x)$ que posee la propiedad*

$$f^*(x) \equiv C \text{ si } f_0 = f_1 = \dots f_m = C, \quad (3.1)$$

donde C es una constante real, decimos que la quasi-interpolación es constante en la reproducción en $[x_0, x_m]$. De esta forma, se tiene que $f^(x)$ es lineal si $f^*(x) = px + q$ como $f_j = px_j + q, j = 0, \dots, m$, para todo $p, q \in \mathbb{R}$.*

De acuerdo a la Definición 1, la idea principal de discretizar el espacio y el tiempo en la Ecuación (1.1), es poder hacer una interpolación de los subespacios generados, de esta forma, para un conjunto de datos iniciales

$$\{(x_j, f_j)\}_{j=0}^m, \quad \text{para } f_j = f(x_j), \quad \text{donde } x_0 < x_1 < \dots < x_m,$$

se puede definir $f^*(x)$ en un intervalo $[x_0, x_m]$ tal que

$$\phi_m(x) = \phi(x) - 2x + x_m + x_0, \quad (3.2)$$

$$\begin{cases} \phi_{-1}(x) = \phi_0(x) + x_0 - x_{-1}, \\ \phi_{m+1}(x) = \phi_m(x) + x_{m+1} - x_m \end{cases} \quad (3.3)$$

y

$$\phi_j(x) = \sqrt{(x - x_j)^2 + c^2}, \quad \text{para } 0 \leq j \leq m-1, c \in \mathbb{R}. \quad (3.4)$$

Así mismo, $f^*(x)$ es una invariante de una quasi-interpolación multi-cuadrática. Para una teoría matemática más detallada y sus correspondientes pruebas, ver [Ro16].

Ahora se presentan las derivadas de forma analítica a partir de la definición de límite.

Definición 2 (Primera derivada mediante diferencias centradas).

$$f'(x_i) = \frac{f(x_i + \Delta x) - f(x_i - \Delta x)}{2\Delta x} + \mathcal{O}_i(\Delta x^2).$$

Definición 3 (Diferencias regresivas).

$$f'(x_i) = \frac{f(x_i) - f(x_i - \Delta x)}{\Delta x} + \mathcal{O}_i(\Delta x).$$

Definición 4 (Diferencias progresivas).

$$f'(x_i) = \frac{f(x_i + \Delta x) - f(x_i)}{\Delta x} - \mathcal{O}_i(\Delta x).$$

Definición 5 (Derivada de orden dos usando diferencias centradas).

$$f''(x_i) = \frac{f(x_i - \Delta x) - 2f(x_i) + f(x_i + \Delta x)}{\Delta x^2} - \frac{\Delta x^2}{12} f^{(4)}(\varepsilon).$$

Definición 6 (Derivada de orden tres usando diferencias centradas).

$$f'''(x_i) = \frac{f(x_i + 2\Delta x) - 2f(x_i + \Delta x) + 2f(x_i - \Delta x) - f(x_i - 2\Delta x)}{2\Delta x^3} + \mathcal{O}_i(\Delta x)^2 \quad (3.5)$$

Dada las Definiciones 2–6, Elementos Finitos nos permite hallar pasos computacionales basado en los pasos que se han realizado anteriormente, de esta forma se tiene la definición de adaptatividad.

Definición 7 (Método Finito Con Pasos Adaptativos). *Los métodos inteligentes, llamados adaptativos, de elementos finitos utilizan información extraída de cálculos anteriores para refinar o modificar la malla localmente para obtener una mejor aproximación de solución u_h . La información necesaria se obtiene utilizando estimaciones de error a posteriori. El objetivo es obtener u_h para ser óptimo en el sentido de que la máxima precisión se logra a un costo computacional mínimo.*

Gracias a la Definición 7, se espera que el error sea pequeño si la malla se encuentra debidamente proporcionada, o si la ecuación diferencial está bien satisfecha por u_h . De hecho, si la solución discreta u_h fuese la solución exacta u , entonces

$$f + u_h'' = 0.$$

Por lo tanto, el elemento residual $\eta_i(u_h)$ es en un sentido proporcional al error e en el elemento I_i . De esta forma, para aumentar la precisión de u_h , es necesario dividir selectivamente los elementos con los residuos más grandes en los más pequeños, ya que esto disminuye h_i y, por lo tanto, también $\eta_i(u_h)$. Por otra parte, al hacerlo de esta manera, es natural esforzarse por una distribución uniforme del error entre los elementos. Esta línea de razonamiento nos lleva a métodos de elementos finitos adaptativos, que controlan automáticamente el error utilizando estimaciones a posteriori en combinación con refinamiento de malla local.

Chapter 4

Propuesta

4.1 Algoritmo Secuencial

Estudio de la ecuación **BBM** a partir del estudio de la estabilización de una modificación de la **KdV**,

$$u_t + \epsilon uu_x + \mu uu_{xxx} = 0, \quad (4.1)$$

propuesta por L. Rosier en el 2016 [Ro16]. Por consecuente, brindar un algoritmo numérico de la Ecuación (4.1), en un espacio rectangular y un espacio periódico usando métodos de diferencias divididas para la comprobación de la Conjetura **PCU**.

Dado que la Conjetura **PCU** surge de forma natural del Análisis Funcional en matemática, como observamos en el Capítulo 3, escogemos el método numérico de diferencias divididas adoptado para la Ecuación (1.1), dado que este se basa en la evolución de funciones sobre un espacio, siguiendo la idea de Xiao, Wang y Zhu [XWZ11].

En esta sección, consideramos la ecuación **KdV**, que es una ecuación no lineal de tercer orden

$$\frac{\partial u(x, t)}{\partial t} + \epsilon u(x, t) \frac{\partial u(x, t)}{\partial x} + u \frac{\partial^3 u(x, t)}{\partial x^3} = 0, \quad (4.2)$$

para un $x \in \Omega = [a, b] \subset \mathbb{R}, t > 0$, con una condición inicial

$$u(x, t) = u^0(x), t = 0, \quad (4.3)$$

y condiciones de frontera

$$x(x, t) = f(t) \quad x \in \partial\Omega, t > 0, \quad (4.4)$$

$$u_x(b, t) = g(t), t > 0. \quad (4.5)$$

Mostramos el análisis numérico para resolver la ecuación **KdV** usando diferencias divididas en un espacio de Banach $\mathcal{L}(\Omega, \Omega)$.

Primero, discretizamos la ecuación

$$u_t + \epsilon uu_x + \mu u_{xxx} = 0, \quad (4.6)$$

en el espacio de tiempo con un paso de τ , donde obtenemos

$$\frac{u_j^{n+1} - u_j^n}{\tau} + \epsilon u_j^n (u_x)_j^n + \mu (u_{xxx})_j^n = 0, \quad (4.7)$$

es decir, para el siguiente paso de la iteración, se realizará

$$u_j^{n+1} = u_j^n - \tau \cdot (\epsilon u_j^n (u_x)_j^n + \mu (u_{xxx})_j^n), \quad (4.8)$$

donde u_j^n es la aproximación del valor $u(x, t)$ en el punto (x_j, t_n) , para $x_j = jh$ y $t_n = n\tau$.

Segundo, discretizamos el espacio para u_x y u_{xxx} , de esta forma, usando las derivadas de la aproximación quasi-interpoladas de u_x , se obtiene

$$(u_x)_j^n = \frac{1}{2} \sum_{k=0}^{m-1} \frac{\phi'_k(x_j) - \phi'_{k+1}(x_j)}{x_{k+1} - x_k} (u_{k+1}^n - u_k^n). \quad (4.9)$$

La quasi-interpolación $f^*(x)$, definido en la Ecuación (3.1), posee una propiedad lineal de tal forma que preserva la monotocidad en $[a, b]$, de esta forma la Ecuación (3.1) se puede volver a escribir como

$$\begin{aligned} f^*(x) &= \frac{1}{2} \sum_{j=1}^{m-1} \left(\frac{\phi_{j+1}(x) - \phi_j(x)}{2(x_{j+1} - x_j)} - \frac{\phi_j(x) - \phi_{j-1}(x)}{2(x_j - x_{j-1})} \right) f_j + \\ &\frac{1}{2} \left(1 + \frac{\phi_1(x) - \phi_0(x)}{(x_1 - x_0)} \right) f_0 + \left(1 - \frac{\phi_j(m) - \phi_{m-1}(x)}{(x_m - x_{m-1})} \right) f_m \end{aligned} \quad (4.10)$$

Además, gracias a las diferencias regresivas, progresivas y centradas del Capítulo 3, tenemos que

$$(f^*(x))' = \frac{1}{2} \sum_{j=0}^{m-1} \frac{\phi'_j(x) - \phi'_{j+1}(x)}{x_{j+1} - x_j} (f_{j+1} - f_j), \quad (4.11)$$

y

$$(f^*(x))'' = \frac{1}{2} \sum_{j=0}^{m-1} \frac{\phi''_j(x) - \phi''_{j+1}(x)}{x_{j+1} - x_j} (f_{j+1} - f_j), \quad (4.12)$$

Tercero, para aproximar u_{xxx} más eficientemente, usamos la discretización de diferencias divididas que van a depender de la Ecuación (4.9), de esta forma obtenemos

$$(u_{xxx})_j^n = \frac{(u_x)_{j+1}^n - 2(u_x)_j^n + (u_x)_{j-1}^n}{h^2}, \quad (4.13)$$

donde $\phi_j(x)$ para $j = 0, \dots, m$ es definido en la Ecuación (3.2) y la Ecuación (3.4).

4.2 Algoritmo Paralelizado

A partir del algoritmo secuencial en la Sección 4.1, se toma en cuenta una cantidad N de ondas para poder procesar grandes volúmenes de estos y poder modelar un océano.

En primer lugar, se distribuye la carga de datos en la inicialización de valores. Sea un intervalo $[A, B]$ con un paso de H , cuya carga de valores iniciales para cada N_i donde $i \in [1, N]$ es de $\frac{B-A}{H}$. De esta forma, se inicializan $\left(\left(\frac{B-A}{H}\right)^2 \times N\right)$ datos en $\mathcal{O}(1)$ para alguna función F . Ahora, para cada N_i evaluada en $F(K_i)$ donde K_i es el valor inicializado para la onda N_i y donde $K \in [A, B]$, se evalúan los pasos u_x y u_{xxx} tal y como en las Ecuaciones (4.9) y (4.13) para cada N_i . De esta forma, el algoritmo es el siguiente.

```
function PARALLELWAVE(result, delta, mu, tao, from, to, step, next, meshes, startIdx,
endIdx)
  o = globalID(0) + startIdx;
  if o < endIdx then
    m = (to[o] - from[o])/step[o]
    start = 1
    result[meshes[o] + 1] = from[o] tmp_a = from[o]
    while start < m do
      result[meshes[o] + start + 1] = function(from[o] + (step[o] * start))
      start + 1
    end while
    start = 1
    result[meshes[o] + m + 1] = 0
    while start < m do
      tmp_a = from[o] + (start * step[o])
      start = start + 1
      val1 = result[meshes[o] + start] * delta[o] * ux(step[o], tmp_a, result, m, next[o], meshes[o])
      val2 = mu[o] * (uxxx(step[o], tmp_a, result, m, next[o], val1, meshes[o]))
      result[meshes[o] + start] = result[meshes[o] + start] - tao[o] * (val1 + val2)
    end while
  end if
end function
```

Al tratar de ejecutar cada parámetro para cada onda, tenemos que tener en cuenta la forma en que estos parámetros están siendo pasados, por lo que es necesario crear una cola para poder controlar la ejecución de ondas. Sea una cola Q , tal que para cada onda N_i , se almacenan los parámetros $\{p_1, p_2, \dots, p_8\}_i$ en Q , de tal forma que en la ejecución de N_i , se lanza la onda $N_{\text{total}-i}$. De esta manera, el algoritmo se ve de la siguiente forma.

```
function QUEUE(waveId, parameters, result)
  if Queue != empty then
    for waveId do
      device_context = runWave(waveId, parameters, results)
    end for
  else
    pushWavesAtQueue()
  end if
end function
```

Chapter 5

Pruebas y Resultados

En este capítulo, se dará ejemplos para la propagación de ondas, donde consideramos la Ecuación (4.6) con $\epsilon = 6$ y $\mu = 1$. La condición inicial es

$$u^0(x) = 12 \left(\frac{3 + 4 \cosh(2x) + \cosh(4x)}{(3 \cosh(x) + \cosh(3x))^2} \right) \quad (5.1)$$

y la solución exacta es

$$u(x, t) = 12 \left(\frac{3 + 4 \cosh(2x - 8t) + \cosh(4x - 64t)}{(3 \cosh(x - 28t) + \cosh(3x - 36t))^2} \right). \quad (5.2)$$

Además, las funciones de frontera $f(t)$ y $g(t)$ pueden ser obtenidas con la solución exacta, considerando un dominio $a \leq x \leq b$ para un $\tau = 0.00001$, $h = 0.1$, y $c = 0.001$, donde dichos datos iniciales y el espacio donde se trabaja son obtenidos de [XWZ11, Sección §3].

5.1 Con pequeñas cantidades de datos, para $a = -5$ y $b = 15$

En esta sección se presentan tres ejemplos de ondas, para tiempos de pasos de $t = 0.01$, $t = 0.05$ y $t = 0.1$. Las pruebas fueron ejecutadas en un computador con un sistema operativo Linux 3.13.0-44-generic Ubuntu x86_64 GNU/Linux con 4GB de RAM core i5, el código implementado fue escrito en MatLab R2014a. La cantidad total de datos utilizados está dado por $\left(\frac{b-a}{h}\right)^2$, cuyo valor en estos ejemplo es de 40 mil datos para cada onda. De esta forma, se obtuvo la siguiente tabla.

	t = 0.01	t = 0.01	t = 0.05	t = 0.05	t = 0.1	t = 0.1
x	Numérica	Exacta	Numérica	Exacta	Numérica	Exacta
-3	0.054469	0.054477	0.039507	0.039507	0.026545	0.026554
-2	0.382932	0.382934	0.275310	0.275315	0.188155	0.188162
-1	2.084132	2.084133	1.390937	1.390946	1.045965	1.045967
0	5.638239	5.638245	2.574822	2.574829	2.000569	2.000572
1	3.192963	3.192964	6.881606	6.881609	1.717096	1.717101
2	0.478487	0.478495	1.207018	1.207024	7.171392	7.171392
3	0.064511	0.064520	0.100947	0.100955	0.464295	0.464299
4	0.008720	0.008723	0.012238	0.012239	0.024306	0.024308
5	0.001178	0.001180	0.001624	0.001630	0.002538	0.002542

Table 5.1: Comparación entre la solución numérica y analítica con $c = 0.001, \tau = 0.00001, h = 0.1$ en $t = 0.01, 0.05, 0.1$.

El tiempo de ejecución de cada onda fue de 1.756835, 1.709990 y 1.733924 segundos, respectivamente. De esta forma, se obtienen las siguientes gráficas.

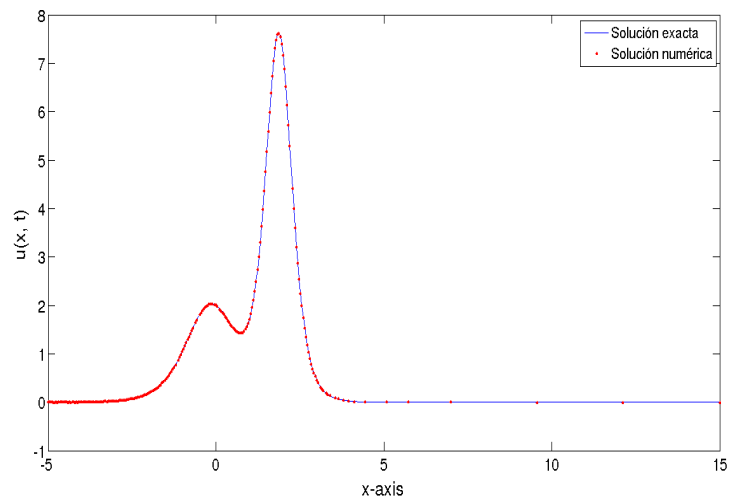


Figure 5.1: Función analítica y numérica del ejemplo con $c = 0.001, \tau = 0.00001, h = 0.1$ en $t = 0.1$

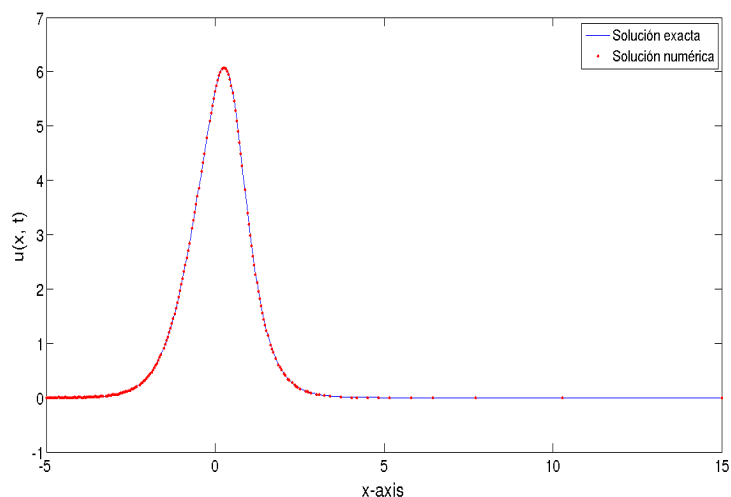


Figure 5.2: Función analítica y numérica del ejemplo con $c = 0.001$, $\tau = 0.00001$, $h = 0.1$ en $t = 0.01$

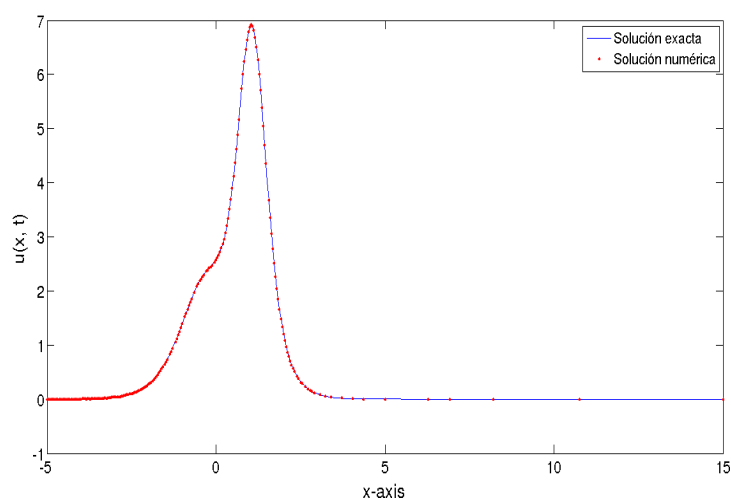


Figure 5.3: Función analítica y numérica del ejemplo con $c = 0.001$, $\tau = 0.00001$, $h = 0.1$ en $t = 0.05$

Las Figuras (5.1), (5.2) y (5.3) representan las posiciones en un tiempo determinado de la onda unidireccional ejercida por la Ecuación (4.6), con una condición inicial de la Ecuación (5.1). De esta forma, se observa que la propagación de la ondas unidireccional coincide con la solución exacta de la Ecuación (5.2) y que se puede predecir la posición de la onda en un tiempo determinado sin la necesidad de pasar por todos los puntos fijos de la solución exacta. Además, se aprecia que la solución numérica es muy aproximada a la solución analítica, y nuestros resultados en la Tabla 5.1 tienen una mejor aproximación a los datos brindados en [XWZ11] con un error de ± 0.0001 en la precisión de resultados.

5.2 Con grandes cantidades de datos, para $a = -100$ y $b = 100$

En esta sección se utilizaron los mismos recursos que en la Sección 5.1. La cantidad de datos utilizados aquí son de 4 millones de datos, para la cual se obtuvo la siguiente tabla.

x	Numérica	Exacta
-2	0.274830	0.275315
-1	1.390190	1.390946
0	2.574214	2.574829
1	6.881440	6.881609
2	1.206954	1.207024
3	0.100736	0.100955
4	0.011516	0.012239
5	0.000841	0.001630
6	0.000194	0.000220

Table 5.2: Comparación entre la solución numérica y analítica con $c = 0.001$, $\tau = 0.00001$, $h = 0.1$ en $t = 0.01, 0.05, 0.1$.

El tiempo de ejecución de la onda fue de 182.694228 segundos y se obtuvo la siguiente gráfica.

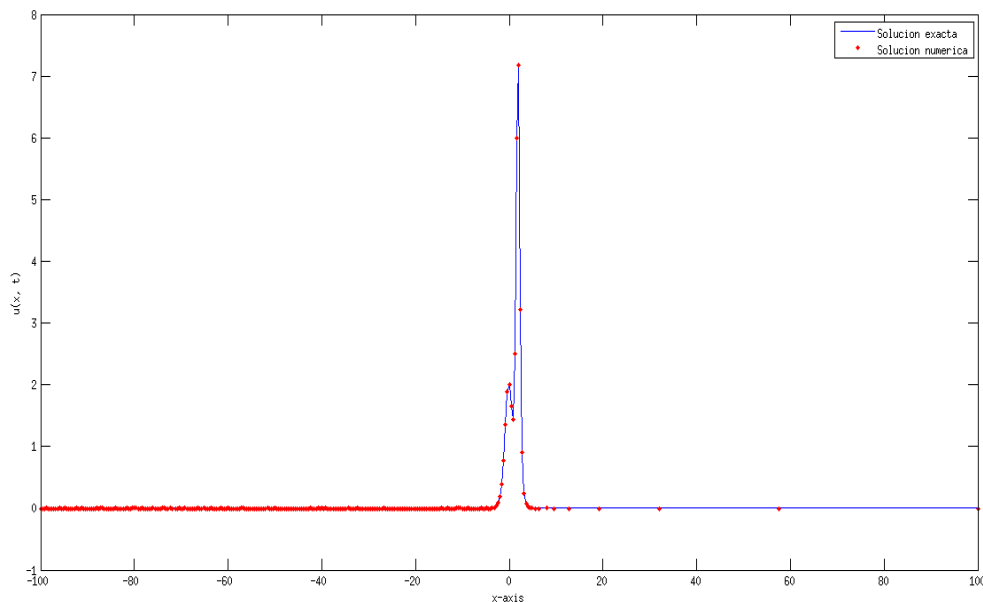


Figure 5.4: Función analítica y numérica del ejemplo con $c = 0.001$, $\tau = 0.00001$, $h = 0.1$ en $t = 0.05$

La Figura (5.4) representa las posiciones de la onda unidireccional ejercida por la

Educación (4.6). Se observa que los primeros computos son los más pesados, y que las predicciones se pueden dar con mayor acierto a partir de la sub-área $[-2, 100]$, esto se debe a que el análisis numérico recién hace uso de las derivadas en la zonas curvas diferentes a 0 de la onda. Además, el margen de error de los datos exactos y numéricos son de ± 0.001 en la precisión de los datos.

5.3 Para 10 ondas simultáneas, con $a = -5$ y $b = 95$

En esta sección, hacemos uso del algoritmo paralelo de la Sección 4.2. Las pruebas fueron ejecutadas en un Linux 4.8.0-59-generic Ubuntu x86_64 GNU/Linux con 8GB de RAM, core i7 y un GPU Nvidia Geforce GT 740M. El código implementado fue escrito en OpenCL 2.0. La cantidad de datos por cada onda es de 1 millón, en total 10 millones de datos. Dado que ya se conoce la convergencia de los resultados por los Ejemplos 5.1 y 5.2, nos enfocamos en realizar un malla para poder simular estas ondas en términos de un océano y cuyas imágenes son las siguientes.

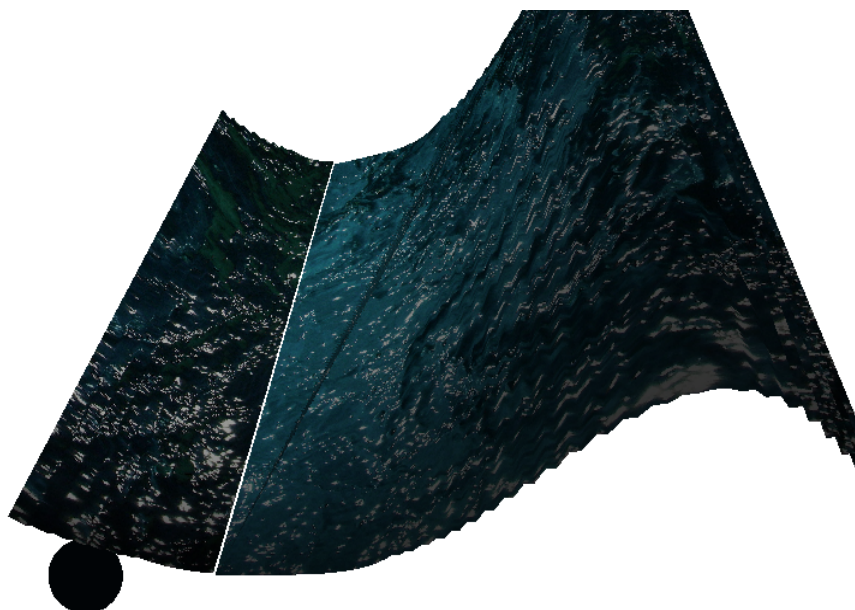


Figure 5.5: Función numérica del ejemplo con $c = 0.001$, $\tau = 0.000001$, $h = 0.1$ en $t = 0.01$, para un total de 10 ondas

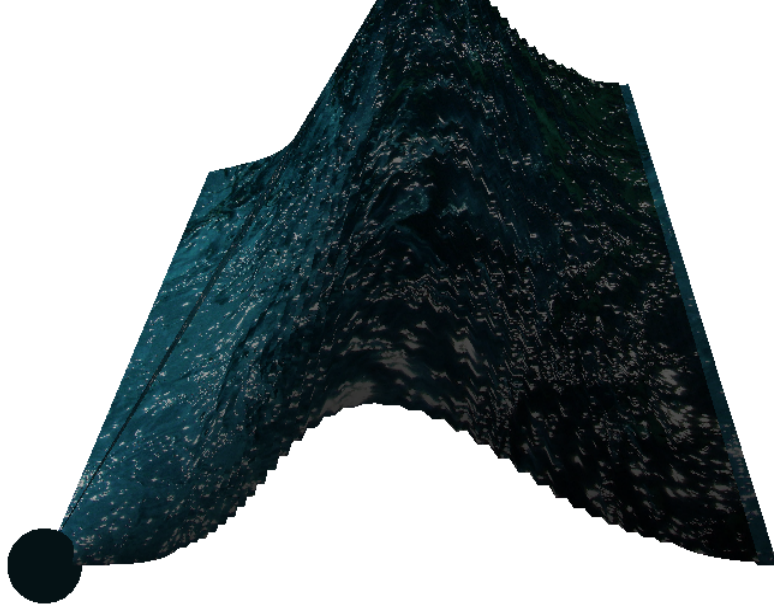
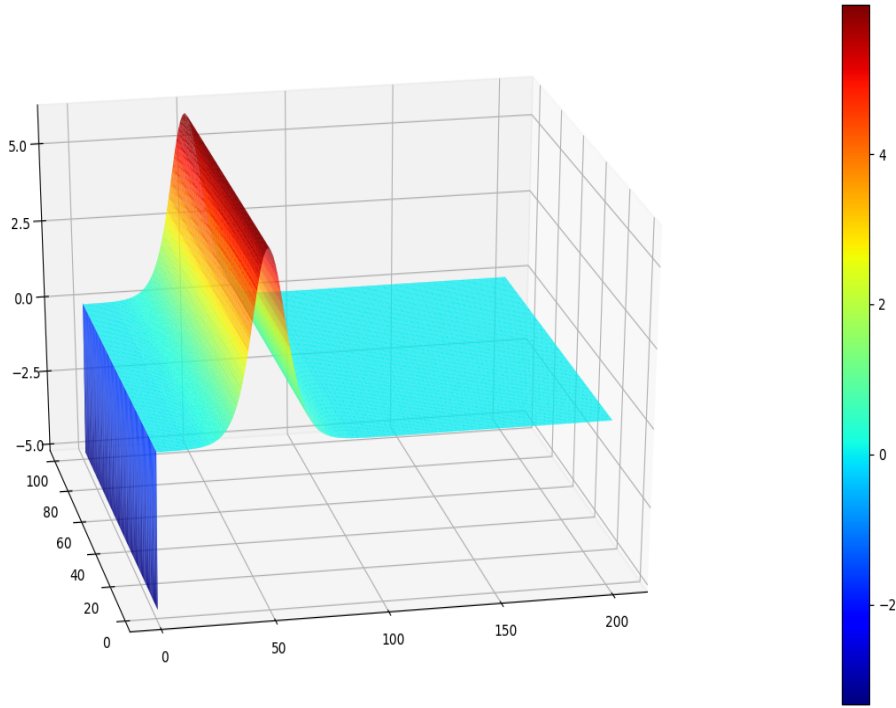


Figure 5.6: Función numérica del ejemplo con $c = 0.001$, $\tau = 0.00001$, $h = 0.1$ en $t = 0.1$, para un total de 10 ondas

Las Figuras (5.5) y (5.6) representan un conjunto de ondas simuladas, cuyos coeficientes de curvatura cambian con el paso del tiempo, las cuales ayudan para realizar predicciones de ondas con un error de precisión de ± 0.0001 .

5.4 Comprobación gráfica del océano real y numérico

Dado que los datos numéricos y reales son muy aproximados, procedemos a obtener una prueba gráfica de los océanos modelados a través de un campo de vectores. Sea O_R el océano con los datos reales y O_N el océano con los datos numéricos de tal forma que vamos a calcular el campo gradiente de vectores de la Figura (5.4) en un tiempo de $t = 50$ cuya imagen modelada se ve de la siguiente forma.

Figure 5.7: O_N detenido en $t = 50$

Ahora, procedemos a hallar el campo de gradiente de vectores para el océano real y que nace del cálculo de la gradiente de la Ecuación (5.2), de esta forma

$$\nabla u_{xt} = \left(\frac{\partial}{\partial x} u_{xt}, \frac{\partial}{\partial t} u_{xt} \right), \quad \text{donde} \quad (5.3)$$

$$\begin{aligned} \frac{\partial}{\partial x} u_{xt} &= \frac{12(4 \sinh(4x - 64t) + 8 \sinh(2x - 8t))}{(\cosh(3x - 36t) + 3 \sinh(x - 28t))^2} - \\ &\frac{24(3 \sinh(3x - 36t) + 3 \sinh(x - 28t))(\cosh(4x - 64t) + 4 \cosh(2x - 8t) + 3)}{(\cosh(3x - 36t) + 3 \cosh(x - 28t))^3} \\ \frac{\partial}{\partial t} u_{xt} &= \frac{12(-64 \sinh(4x - 64t) - 32 \sinh(2x - 8t))}{(\cosh(3x - 36t) + 3 \cosh(x - 28t))^2} - \\ &\frac{(24(-36 \sinh(3x - 36t) - 84 \sinh(x - 28t))(\cosh(4x - 64t) + 4 \cosh(2x - 8t) + 3))}{(\cosh(3x - 36t) + 3 \cosh(x - 28t))^3} \end{aligned} \quad (5.4)$$

En el caso de O_N creamos un campo de vectores eléctricos de forma separada, es decir, hacemos el cálculo de la energía basada en la definición de la Ecuación (4.10) con el fin de obtener una suma de los datos numéricos con respecto las acumulaciones de

las derivadas. Ahora, graficamos ambos campos vectoriales de la zona superior de los océanos, ya que es la que nos interesa por tener el cambio de curvatura.

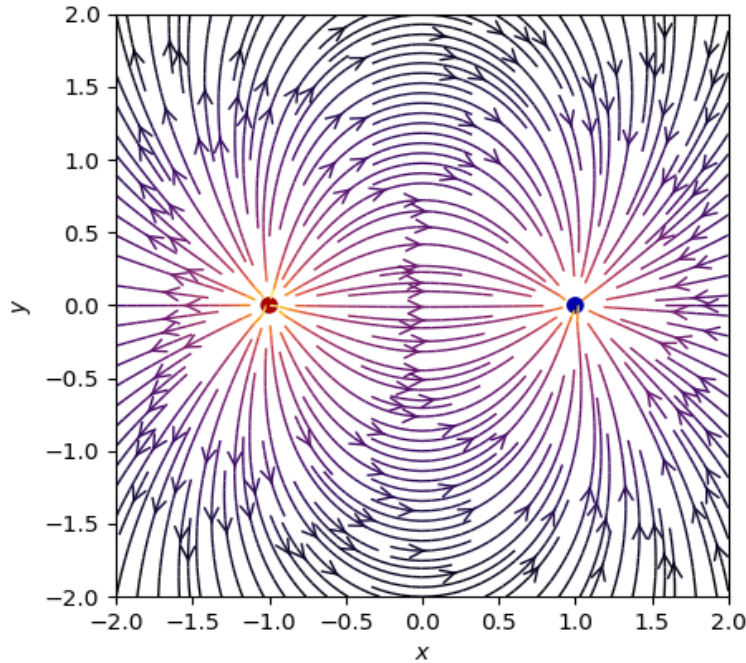


Figure 5.8: Campo de vectores de $O_{\mathcal{R}}$ y $O_{\mathcal{N}}$

La representación de estos campos de vectores nos demuestran que gráficamente los cambios de curvaturas de $O_{\mathcal{R}}$ y $O_{\mathcal{N}}$ son muy cercanos, lo cuál respaldan una vez más nuestros datos numéricos. Los experimentos computacionales de esta Sección fueron desarrollados en el Centro de Alto Rendimiento Computacional de la Amazonía Peruana del Instituto de Investigaciones de la Amazonía Peruana. Mayor información en: <http://iiap.org.pe/manati>

5.5 Para 1000 ondas simultáneas, con $a = -100$ y $b = 100$

Ahora que ya comprobamos en la Sección 5.4 que las curvas y picos generados por nuestro océano numérico y nuestro océano real son muy aproximados, en esta sección hacemos uso del algoritmo paralelo de la Sección 4.2 con la finalidad de crear un océano más *caótico*, esto lo realizamos aumentando el número de ondas a 1000. Las pruebas fueron ejecutadas en un Linux 4.8.0-59-generic Ubuntu x86_64 GNU/Linux con 8GB de RAM, core i7 y un GPU Nvidia Geforce GT 740M. El código implementado fue escrito en OpenCL 2.0. La cantidad de datos por cada onda es de 20 millones, en total 2 billones de datos. Dado que ya se conoce la convergencia de los resultados por los Ejemplos 5.1 y 5.2, nos enfocamos

en realizar un malla para poder simular estas ondas en términos de un océano y cuya imagen es la siguiente.



Figure 5.9: Función numérica del ejemplo con $c = 0.001$, $\tau = 0.00001$, $h = 0.1$ en $t = 0.1$, para un total de 1000 ondas

En la Figura (5.9) se puede apreciar que los bordes no son tan suaves como las Figuras (5.5) y (5.6), esto se debe a que obtuvimos un modelaje más real al aumentar a 1000 el número de ondas, así mismo se ven una mayor variedad en las alturas a lo largo del océano que nos indica que las fluctuaciones se han vuelto visualmente más caóticas.

Chapter 6

Conclusiones y Trabajos Futuros

El uso de diferencias divididas que parte del algoritmo **MEF**, obtiene una buena discretización de orden $\mathcal{O}(h^2)$ gracias a su truncamiento cuadrático h^2 sin la necesidad de aumentar el orden de este, gracias a su comportamiento natural de onda en el intervalo $[a, b]$ para $a = 0$ y $b = L$ donde L es una constante real, haciendo que la **KdV** pueda brindar datos aproximados a su solución exacta sin pérdida de mucha información en la evolución del tiempo, como hemos podido ver en el ejemplo del Capítulo 3. El aumento del orden del truncamiento (e.g., orden tres) para este análisis numérico en particular no es necesario, dado que sólo conseguiríamos aumentar el tiempo de cálculo, pero no obtener una mejor aproximación de datos como se puede ver en la Ecuación (3.5). En efecto, el análisis gráfico de la Sección 5.4 y el modelaje del océano numérico con 1000 ondas y presencia de *caos* de la Sección 5.5, nos confirman que las predicciones realizadas por la Ecuación (4.1) en las Tablas 5.1 y 5.2 son válidas y correctas.

6.1 Problemas encontrados

- La discretizaciones de las Ecuaciones (4.10), (4.11) y (4.12), resultaban computacionalmente muy pesadas en número de iteraciones dependientes, por lo que se reparametrizaron las ecuaciones en base al método de diferencia divididas, de esta forma se obtuvo la Ecuación (4.13).
- La paralelización del algoritmo no funcionó a los inicios para un mal manejo de la cola del propio OpenCL 2.0, por lo que se tuvo que crear una cola y decir el orden en que los parámetros van a ser enviados para cada onda.
- La falta de recursos computacionales no le permitió al autor realizar el modelaje de un océano con 1 millón de ondas, que hubiera sido un caso bastante interesante de presentar en cuestión de las predicciones de ondas.

6.2 Recomendaciones

Gracias al análisis numérico propuesto en Capítulo 4, se podría extender en un futuro el análisis de la parte lineal de la ecuación **BBM** en un espacio periódico (e.g. Torus). Así mismo, se puede aplicar estudios de predicción de ondas en base a los parámetros iniciales de la ecuación propuesta por L. Rosier, gracias a la convergencia de funciones presentada en el Capítulo 3.

6.3 Trabajos futuros

Realizar la discretización de la Ecuación (4.1) en un espacio periódico (e.g. Torus), con el objetivo de disminuir el nivel de *libertad* de la onda.

Apéndice

Definiciones

Definición 8 (Espacio de Banach). *Un espacio vector normado es un par $(X, \|\cdot\|)$, que consiste de un espacio vector real X y una función $X \rightarrow \mathbb{R} : x \mapsto \|x\|$, de tal forma que satisfice las siguientes propiedades:*

(N1) $\|x\| \geq 0$ para todo $x \in X$, con igualdad sí y sólo sí $x = 0$.

(N2) $\|\lambda x\| = |\lambda| \|x\|$ para todo $x \in X$ y $\lambda \in \mathbb{R}$.

(N3) $\|x + y\| \geq \|x\| + \|y\|$ para todo $x, y \in X$.

Sea $(X, \|\cdot\|)$ un espacio vector normado. Entonces la fórmula

$$d(x, y) := \|x - y\|,$$

para $x, y \in X$, define una función de distancia en X . La topología resultante es denotado por $\mathcal{U}(X, \|\cdot\|) := \mathcal{U}(X, d)$. X es llamado un **espacio de Banach** si la métrica del espacio (X, d) es **completo** (i.e., cada secuencia Cauchy en X converge).

Definición 9 (Espacio de Banach \mathcal{L}^p). Sea (M, \mathcal{A}, μ) un espacio de medida (i.e., M es un conjunto, $\mathcal{A} \subset 2^M$ es un σ -álgebra y $\mu : \mathcal{A} \rightarrow [0, \infty]$ es una medida). Fija una constante $1 \leq p < \infty$. Una función medible $f : M \rightarrow \mathbb{R}$ es llamada p -integrable si $\int_M |f|^p d\mu < \infty$ y el espacio de funciones p -integrable en M será denotado por

$$\mathcal{L}^p(\mu) := \left\{ f : M \rightarrow \mathbb{R} \mid f \text{ es medible y } \int_M |f|^p d\mu < \infty \right\}.$$

La función $\mathcal{L}^p \rightarrow \mathbb{R} : f \mapsto \|f\|_p$ es definido por

$$\|f\|_p = \left(\int_M |f|^p \right)^{1/p},$$

es no negativo y satisfice la desigualdad triangular (i.e., inigualdad de Minkowski).

Definición 10 (Elemento Finitos). Entre los distintos métodos utilizados para discretizar un problema, se refiere a aquellos que emplean distintas funciones de prueba para materializar la aproximación y que se conocen como **elementos finitos**. Se utiliza las siguientes propiedades:

(P1) Representación precisa de la geometría compleja.

(P2) Inclusión de propiedades de material disímiles.

(P3) Fácil representación de la solución total.

(P4) Captura de efectos locales.

En el Capítulo 4, se utiliza principalmente la propiedad (P4) y el uso del Teorema 2 para la captura de las propiedades de convergencia de secuencias y después mapearlo al espacio total, de esta forma se toma las diferencias discretas de las Ecuaciones en el Capítulo 3 para poder discretizar las ondas largas con pequeñas amplitudes en el espacio total.

Teoremas

Conjetura 1 (Propiedad de Continuación Única). *Existe un número $\delta > 0$ tal que para cualquier $v_0 \in H^1(\mathbb{T})$ con $\|v_0\|_{H^1(\mathbb{T})} < \delta$, si la solución $v = v(x, t)$ de*

$$\left\{ \begin{array}{ll} v_t - v_{xxt} + v_x + vv_x = 0, & \text{para } x \in \mathbb{T}, \\ v(x, 0) = v_0(x), & \text{para } x \in \mathbb{T} \end{array} \right\},$$

satisfice

$$v(x, t) = 0 \quad \forall (x, t) \in \omega \times (0, T),$$

para algún conjunto abierto no vacío $\omega \subset \mathbb{T}$ and algún $T > 0$, entonces $v_0 = 0$ (y así mismo $v \equiv 0$).

La conclusión de que todas las trayectorias que emanan de datos cercanos al origen en la ecuación **BBM**, que son efectivamente atraídas por el origen, es válida siempre que se cumpla la Conjetura 1. De esta forma, se toma la **PCU** para la Ecuación (4.1) y el espacio seteado en las Ecuaciones (4.3), (4.4) y (4.5), en el cual se prueba la estabilización siempre y cuando que las trayectorias (e.g. ondas largas con pequeñas amplitudes unidireccionales) convergan en la frontera con los datos de valores iniciales. Después, se confirma la estabilización para la ecuación **KdV** y se trabaja para la ecuación **BBM** el mismo espacio seteado que en el análisis numérico presente, de esta forma se obtiene un **margen de error de los datos de la estabilización** en la **BBM** desde la estabilización de la Ecuación (4.1).

Teorema 1 (Caracterización de espacios compactos). *Sea (X, d) un espacio métrico y sea $K \subset X$. Entonces, los siguientes enunciados son todos equivalentes.*

- (i) K es secuencialmente compacto.
- (ii) K es completo y totalmente limitado.
- (iii) Cada cover abierto de K tiene un finito subcover.

El espacio de Banach $\mathcal{L}(\Omega, \Omega)$, utiliza el Teorema 1 para poder establecer las condiciones de convergencia de las funciones en las Ecuaciones (4.4) y (4.5). Dado que Ω es un espacio compacto, se toma la convergencia de las series de funciones en el borde del espacio, para poder limitar las convergencias de las ondas en \mathcal{L} .

Teorema 2 (Hahn–Banach). *Sea X un espacio vector normado y sea $\rho : X \rightarrow \mathbb{R}$ una quasi-norma. Sea $Y \subset X$ un sub-espacio lineal y sea $\phi : Y \rightarrow \mathbb{R}$ tal que $\phi(x) \leq \rho(x)$ para todo $x \in Y$. Entonces, existe una función lineal $\Phi : X \rightarrow \mathbb{R}$ tal que*

$$\Phi|_Y = \phi, \quad \Phi(x) \leq \rho(x) \quad \text{para todo } x \in X.$$

Dado un intervalo $[x_0, x_m]$ para las funciones de la Ecuación (3.1), se toma un sub-espacio $[x_0, x_{m_i}]$ de tal forma que se hace un mapeamiento de las funciones del primer sub-espacio al segundo sub-espacio, de esta forma se utiliza el Teorema 2 y así extender las propiedades de convergencia de las ondas largas con pequeñas amplitudes para la siguientes iteraciones del algoritmo.

Bibliography

- [AbSe81] M.J. Ablowitz, H. Segur, *Solitons and the Inverse Scattering Transform*. SIAM, Philadelphia, 1981.
- [BaTu16] S. Battal Gazi Karakoc and Turgut Ak, *Numerical solution of Rosenau-KdV equation using subdomain finite element method*. New Trends in Mathematical Sciences. **1** (2016), 223-235.
- [BBM72] T. Benjamin, J. Bona, J. and J. Mahony, *Model equations for long waves in nonlinear, dispersive media*. Philos. Trans. Royal Soc. London Series A **272** (1972), 47-78.
- [Bh79] P.L. Bhatnagar, *Nonlinear Waves in One-Dimensional Dispersive Systems*. Clarendon Press. Oxford, 1979.
- [BoTz09] J.L. Bona and N. Tzvetkov, *Sharp well-posedness for the BBM equation*. Discrete Contin. Dyn. Syst. **23** (4) (2009) 1241-1252.
- [CeSe02] E. Cerezo and F.J. Serón, *Rendering Natural Waters*. Springer-Verlag London (2002) 481-498.
- [DEGM84] R.K. Dodd, J.C. Eilbek, J.D. Gibbon, H.C. Morris, *Solitons and Nonlinear Wave Equations*. Academic Press. London, 1984.
- [Dr83] P.G. Drazin, *Solitons*. Cambridge University Press. Cambridge, London, 1983.
- [Ga74] G.S. Gardner, *The KdV equation and generalizations, methods for exact solutions*. Commun. Pure Appl. Math. **27** (1974) 97-133.
- [GGKM67] G.S. Gardner, J.M. Greene, M.D. Kruskal, R.M. Miura, *A method for solving the KdV equation*. Phys. Rev. Lett. **19** (1967) 1095-1097.
- [KiLe99] Y. D. Kim and H. Y. Lee, *The convergence of finite element Galerkin solution for the Rosenau equation*. The Korean Journal of Computational and Applied Mathematics, **6** (1999), 1-13.
- [KRD12] S. Kapoor, S. Rawat and S. Dhawan, *Numerical Investigation of Separated Solitary Waves Solution for KDV Equation through Finite Element Technique*. International Journal of Computer Applications. **40** (2012), 27-33.
- [La68] P.D. Lax, *Integrals of nonlinear equations of evolution and solitary waves*. Commun. Pure Appl. Math. **11** (1968) 167-190.

-
- [La80] G.L. Lamb, *Elements of Soliton Theory*. Wiley. New York, 1980.
- [LaTh03] S. Larsson and V. Thomee, *Partial Differential Equations with Numerical Methods (Texts in Applied Mathematics)*. Springer Science & Business Media (2003)
- [LBR07] N. Lee, N. Baek and K. W. Ryu, *Real-Time Simulation of Surface Gravity Ocean Waves Based on the TMA Spectrum*. Computational Science - ICCS (2007) 122-129.
- [LRZ10] C. Laurent, L. Rosier and B.-Y. Zhang, *Control and Stabilization of the Korteweg-de Vries equation on a periodic Domain*. Comm. in Partial Diff. Eq. **35** (2010), 707-744
- [Mi01] S. Micu, *On the controllability of the linearized Benjamin-Bona-Mahony equation*. SIAM J. Cont. Optim **39** (2001), 1677-1696.
- [PaRo08] A. F. Pazoto and L. Rosier, *Stabilization of a Boussinesq system of KdV-KdV type*. Systems Control Lett. **57** (2008), 595-601.
- [Ro10] D. Roumégoux, *A symplectic non-squeezing theorem for BBM equation*. Dyn. Partial Differ. Equ. **7** (4) (2010) 289-305.
- [Ro16] L. Rosier, *On the Benjamin-Bona-Mahony equation with a localized damping*. Journal Math. **49** Study (2016), 195-204
- [Ro97] L. Rosier, *Exact boundary controllability for the Korteweg-de Vries equation on a bounded domain*. ESAIM Control Optim. Calc. Var. **2** (1997), 33-55.
- [RoZh12] L. Rosier and B.-Y. Zhang, *Unique continuation property and control for the Benjamin-Bona-Mahony equation on the torus*. Preprint 2012, 1202.2667.
- [Vr94] G. de Vries, *Bijdrage tot de Kennis der Lange Golven*. Acad. Proefschrift, Universiteit van Amsterdam. National Library of the Netherlands, Loosjes 1894.
- [Wh74] G.B. Whitham, *Linear and Nonlinear Waves*. Wiley. New York, 1974.
- [XWZ11] M. L. Xiao, R. H. Wang and C. G. Zhu, *Applying Multiquadric Quasi-Interpolation to Solve KdV Equation*. Journal of Mathematical Research & Exposition (2011) 191-201.
- [YiPi13] Yong-Xue Ying and Guang-Ri Piao, *Quadratic B-Spline finite element method for the Benjamin-Bona-Mahony-Burgers equation*. East Asian Mathematical Journal. **29** (2013), 503-510.
- [ZaKr65] N.J. Zabusky, M.D. Kruskal, *N.J. Zabusky, M.D. Kruskal, Interaction of solitons in collisionless plasma and the recurrence of initial states*. Phys. Rev. Lett. **15** (1965) 240-243.
- [ZaSh72] V. E Zakharov and A. B. Shabat, *Exact theory of two-dimensional self-focusing and one-dimensional self-modulation of Waves in nonlinear media*. Soviet Phys. JETP. **34** (1972), 62-69.
- [ZhZu03] X. Zhang and E. Zuazua, *Unique continuation for the linearized Benjamin-Bona-Mahony equation with space-dependent potential*. Math. Ann. **325** (2003), 543-582.
-