

# Etude du Soliton des équations de Korteweg de Vries et de Schrödinger non linéaire

Lucas Palazzolo

Strasbourg  
UFR de Mathématiques et d'Informatique  
Tutrice de Mémoire : Clémentine Courtès  
26 août 2022



# Table des matières

<b>Remerciements</b>	<b>4</b>
<b>Introduction</b>	<b>6</b>
<b>Rappels</b>	<b>8</b>
EDP dispersive . . . . .	8
Méthode de Heun . . . . .	8
Méthode de RK4 . . . . .	9
Méthode de Newton-Raphson . . . . .	9
Identité géométrique . . . . .	9
<b>1 Analyse de l'équation de Korteweg de Vries</b>	<b>10</b>
1.1 Propriétés vérifiées par l'équation . . . . .	10
1.1.1 L'équation de KdV est une EDP dispersive et non linéaire . . . . .	10
1.1.2 Conservation de la norme $L^2$ . . . . .	11
1.1.3 Conservation de l'Hamiltonien . . . . .	12
1.1.4 Invariance d'échelle . . . . .	13
1.1.5 Translation espace-temps . . . . .	13
1.2 Propriétés vérifiées par le soliton . . . . .	13
1.2.1 Résolution de l'équation . . . . .	13
1.2.2 Portrait de phase . . . . .	16
<b>2 Analyse de l'équation de Schrödinger non linéaire</b>	<b>18</b>
2.1 Propriétés vérifiées par l'équation . . . . .	18
2.1.1 L'équation de NLS est une EDP dispersive et non linéaire . . . . .	18
2.1.2 Conservation de la norme $L^2$ . . . . .	19
2.1.3 Conservation du Moment . . . . .	19
2.1.4 Conservation de l'Energie . . . . .	21
2.1.5 Invariance d'échelle . . . . .	22
2.1.6 Invariance de phase . . . . .	22
2.1.7 Translation espace-temps . . . . .	23
2.2 Propriétés vérifiées par le soliton . . . . .	23
2.2.1 Résolution de l'équation . . . . .	23
2.2.2 Portrait de phase . . . . .	26
<b>3 Résolution numérique des deux équations</b>	<b>28</b>
3.1 Résolution de l'équation de KdV par un schéma compact symplectique . . . . .	28
3.1.1 L'opérateur des différences finies compact . . . . .	28
3.1.2 Conservation de l'hamiltonien discret . . . . .	29
3.1.3 Implémentation du schéma . . . . .	32
3.2 Résolution de l'équation de NLS par la méthode de tir . . . . .	34
3.2.1 Introduction . . . . .	34
3.2.2 Au tour de l'équation de NLS . . . . .	35
3.3 Adaptation des schémas à l'autre équation . . . . .	38
3.3.1 Méthode de tir pour l'équation de KdV . . . . .	38
3.3.2 Schéma compact pour l'équation de NLS . . . . .	40
3.3.2.1 Calcul de la variation de la densité de l'énergie de NLS . . . . .	40

3.3.2.2	Discrétisation de la variation de l'énergie . . . . .	42
3.3.2.3	Consistance de $\left(\frac{\delta \tilde{E}}{\delta a}\right)_j^n$ et $\left(\frac{\delta \tilde{E}}{\delta b}\right)_j^n$ . . . . .	44
3.3.2.4	Schéma compact . . . . .	44
3.3.2.5	Conservation de l'énergie discrète . . . . .	48
3.3.2.6	Conservation de la norme $L^2$ . . . . .	48
3.3.2.7	Conservation du moment . . . . .	49
<b>4</b>	<b>Conclusion</b>	<b>50</b>
<b>A</b>	<b>Résolution de l'équation de NLS en posant <math>u(t, x) = \phi(x)e^{i\omega t}</math></b>	<b>52</b>
<b>B</b>	<b>Codes</b>	<b>53</b>
B.1	Méthode de Newton . . . . .	54
B.2	Equation de KdV . . . . .	54
B.2.1	Portrait de phase . . . . .	54
B.2.2	Schéma compact . . . . .	54
B.2.3	Méthode de Heun pour le schéma compact . . . . .	57
B.2.4	Méthode de tir pour KdV . . . . .	58
B.3	L'équation de NLS . . . . .	59
B.3.1	Portrait de phase . . . . .	59
B.3.2	Méthode de tir . . . . .	60
B.3.3	Schéma compact pour l'équation de NLS . . . . .	61
	<b>Références</b>	<b>66</b>

## Remerciements

C'est avec ces derniers mots que je termine cette merveilleuse petite aventure qu'était ce mémoire. Je tiens tout particulièrement à remercier Madame Clémentine Courtès avec qui j'ai découvert le calcul scientifique et qui m'a soutenu et répondu à toutes les diverses questions que j'ai pu poser. Merci de m'avoir fait découvrir ce qu'est le monde de la recherche.

Je voudrais remercier pour finir mes parents et mon frère qui ont su m'écouter et qui ont essayé tant bien que mal de comprendre ce que j'étais entrain de faire. Merci à vous de m'avoir toujours poussé à donner le meilleur de moi même...



## Introduction

Un soliton est une onde solitaire, c'est-à-dire une onde localisée spatialement, qui se propage sans se déformer dans un milieu non linéaire et dispersif. En 1834, un ingénieur hydrodynamicien, John Scott Russell, a observé ce phénomène en se promenant le long d'un canal. Fasciné par cette observation, il consacre dix années de sa vie à étudier le phénomène. Il fallut attendre 1871, année où le mathématicien et physicien français Joseph Boussinesq publia la première théorie mathématique décrivant l'observation de Russell. C'est en 1895 qu'une équation a été proposée pour comprendre les solitons et leurs propriétés [10]. Korteweg est l'un des premiers chercheurs à avoir découvert des ondes solitaires qui modélisent des ondes exceptionnelles se propageant à la surface de l'eau sur de très longues distances sans déformation.

Les solitons sont au coeur de la modélisation de nombreux autres phénomènes physiques comme en optique où ils ont permis de comprendre comment générer une vague de lumière qui se propage sans déformation (une impulsion électromagnétique), c'est-à-dire un soliton optique. Ainsi les scientifiques se sont servis de ces solitons pour fabriquer des lasers de haute intensité. Ils ont aussi servi pour la formation de faisceaux laser en optique non linéaire modélisée par l'équation de Schrödinger non linéaire [10].

Un célèbre exemple de solitons est illustré par les mascarets. Les mascarets sont des vagues formées par la marée et remontant les fleuves et les estuaires. Ils se terminent généralement en amont en un soliton. Selon plusieurs rumeurs, Léopoldine, la fille de Victor Hugo, aurait été emportée par un mascaret sur la Seine [1].

L'objet de ce mémoire est de traiter le soliton de l'équation de Korteweg de Vries, le soliton de l'équation de Schrödinger non linéaire théoriquement et numériquement à l'aide de différentes méthodes permettant de capter les solitons et certaines de leurs propriétés fondamentales.





## Rappels

Dans cette partie, nous ferons des rappels succincts sur les différentes notions abordées dans le mémoire. Le but étant seulement de simplifier la compréhension des différents sujets abordés pour le lecteur. Pour des rappels plus rigoureux, il conviendra d'aller voir les diverses sources citées ci-après.

### EDP dispersive

Pour montrer qu'une équation aux dérivées partielles (par la suite EDP) est dispersive ou non, il faut tout d'abord commencer par linéariser notre équation. Pour cela prenons l'exemple de l'EDP ci-dessous :

$$\partial_t u + \partial_x f(u) + \partial_x^2 g(u) = 0.$$

Lorsque l'on linéarise, il faut choisir un état d'équilibre autour duquel nous allons travailler. Pour cela notons  $U$  l'état d'équilibre (indépendant du temps et de l'espace) et  $u^*$  une petite variation autour de cet état.

$$u(t, x) = U + u^*(t, x).$$

Maintenant il ne reste plus qu'à injecter notre solution dans l'équation en gardant seulement les termes d'ordre 1 en  $u^*$  et en n'oubliant pas que  $U$  est une constante.

$$\begin{aligned} \partial_t(U + u^*) + \partial_x f(U + u^*) + \partial_x^2 g(U + u^*) &= 0 \\ \partial_t u^* + \partial_x (f(U) + f'(U)u^* + O(u^{*2})) + \partial_x^2 (g(U) + g'(U)u^* + O(u^{*2})) &= 0 \\ \partial_t u^* + f'(U)\partial_x u^* + g'(U)\partial_x^2 u^* + O(u^{*2}) &= 0 \end{aligned}$$

Une EDP est dispersive si des ondes planes monochromatiques de fréquences différentes se propagent à des vitesses différentes. Ici on se place dans le cas d'ondes planes monochromatiques, de la forme :

$$u^*(t, x) = e^{i(\langle \xi, x \rangle - \omega t)} \begin{cases} u & : \text{solution de l'EDP} \\ \xi & : \text{vecteur d'onde} \\ \omega & : \text{pulsation temporelle.} \end{cases}$$

Pour montrer que notre EDP est dispersive, il faut à partir de la relation de dispersion  $\omega = W(\xi)$  (obtenue en remplaçant  $u$  dans l'EDP linéarisée) montrer que  $\det(\text{Hess}(W)) \neq 0$ . Dans notre cas, on travaille en une seule dimension donc il faut que  $W'' \neq 0$ . Dans le registre physicien, on parle plutôt de vitesse de groupe  $v_g(\xi) = W'(\xi)$  et de vitesse de phase  $v_\varphi(\xi) = \frac{W(\xi)}{\xi}$  pour montrer la dispersion. En effet, on a la propriété équivalente suivante : si la vitesse de phase et la vitesse de groupe sont différentes alors l'EDP est dispersive. Pour des explications plus rigoureuses, le cours de Sylvie Benzonie [5] convient parfaitement.

### Méthode de Heun

Prenons comme exemple l'équation différentielle ordinaire (par la suite EDO) suivante [6] :

$$\begin{cases} u'(t) = f(t, u(t)) & \text{avec } t \in [t_0, t_0 + T], \\ u(t_0) = u_0 \end{cases}$$

où l'on subdivise  $[t_0, t_0 + T]$  en  $N$  sous intervalles  $[t^n, t^{n+1}]$ . On définit le pas de discrétisation  $\Delta t = T/N$  et  $v^n$  la valeur approchée de  $u(t^n)$ . Le schéma de Heun est un schéma d'ordre 2

obtenue en approximant l'intégrale de la fonction  $f$  par la méthode des trapèzes. Le schéma est défini par :

$$v^{n+1} = v^n + \frac{\Delta t}{2}k_1 + \frac{\Delta t}{2}k_2$$

$$\text{où } \begin{cases} v^0 = u_0, \\ k_1 = f(t^n, v^n), \\ k_2 = f(t^n + \Delta t, v^n + \Delta t k_1). \end{cases}$$

## Méthode de RK4

Prenons la même EDO et la même discrétisation que le schéma de Heun. La méthode de RK4 (Runge-Kutta) est un schéma d'ordre 4 obtenu en approximant l'intégrale de la fonction  $f$  par la méthode de Simpson [6]. Le schéma est défini par :

$$v^{n+1} = v^n + \frac{\Delta t}{6}k_1 + \frac{\Delta t}{3}k_2 + \frac{\Delta t}{3}k_3 + \frac{\Delta t}{6}k_4$$

$$\text{où } \begin{cases} v^0 = u_0, \\ k_1 = f(t^n, v^n), \\ k_2 = f(t^n + \frac{\Delta t}{2}, v^n + \frac{\Delta t}{2}k_1), \\ k_3 = f(t^n + \frac{\Delta t}{2}, v^n + \frac{\Delta t}{2}k_2), \\ k_4 = f(t^n + \Delta t, v^n + \Delta t k_3). \end{cases}$$

## Méthode de Newton-Raphson

La méthode de Newton-Raphson [2] consiste à approximer précisément un zéro de la fonction étudiée. De base, cela s'applique aux fonctions réelles. Dans notre cas, on recherche un zéro d'une application à valeurs réelles. Pour cela, nous utiliserons le cas généralisé de la méthode de Newton-Raphson. Notons  $F$  l'application étudiée et  $J$  sa jacobienne. Ainsi, nous avons le schéma suivant dont on peut choisir comme critère d'arrêt  $|F(x_{k+1})| < \varepsilon$  où  $\varepsilon$  est la précision de notre approximation.

$$x_{k+1} = x_k - J^{-1}(x_k)F(x_k)$$

où  $x_k$  est l'approximation du zéro de  $F$  à l'itération  $k$ .

## Identité géométrique

Soient  $a$  et  $b$  deux réels. Pour tout entier  $n \in \mathbb{N}$ ,

$$a^{n+1} - b^{n+1} = (a - b) \sum_{k=0}^n a^k b^{n-k}$$

Pour démontrer cette formule, il suffit de faire une simple récurrence.

# 1 Analyse de l'équation de Korteweg de Vries

L'équation de Korteweg de Vries représente un modèle simple d'équation d'hydrodynamique possédant un soliton dans sa version normalisée.

$$\partial_t u(t, x) + u(t, x) \partial_x u(t, x) + \partial_x^3 u(t, x) = 0 \quad \forall (t, x) \in \mathbb{R}_+ \times \mathbb{R} \quad (\text{KdV})$$

Dans le cas de (KdV),  $u(t, x)$  modélise la hauteur de la surface de l'eau par rapport à un niveau de référencement. On peut donc supposer que  $u(t, x) > 0 \quad \forall (t, x) \in \mathbb{R}_+ \times \mathbb{R}$  et que la solution est localisée, c'est-à-dire que la solution et ses dérivées tendent vers 0 quand  $x$  tend vers l'infini.

## 1.1 Propriétés vérifiées par l'équation

Dans ce paragraphe, nous allons étudier les différentes propriétés vérifiées par (KdV). En effet, l'équation de (KdV) satisfait plusieurs propriétés indépendamment de la solution recherchée. C'est-à-dire que peu importe la solution étudiée, les propriétés de l'équation seront satisfaites.

### 1.1.1 L'équation de KdV est une EDP dispersive et non linéaire

Tout d'abord, on peut distinguer des termes linéaires  $\partial_t u$ ,  $\partial_x^3 u$  et des termes non linéaires  $u \partial_x u$  dans (KdV). Comme nous l'avons vu dans l'introduction, il faut d'abord linéariser notre équation pour étudier sa dispersion. Pour cela, nous allons commencer par réécrire notre équation.

$$\partial_t u(t, x) + \frac{1}{2} \partial_x (u^2(t, x)) + \partial_x^3 u(t, x) = 0 \quad \forall (t, x) \in \mathbb{R}_+ \times \mathbb{R} \quad (1.1)$$

Maintenant nous allons linéariser (1.1) autour de l'état d'équilibre  $U = 0$ ,

$$u(t, x) = U + u^*(t, x) \quad (1.2)$$

où  $u^*$  est une petite perturbation. Remplaçons  $u$  par (1.2) dans (1.1) en n'oubliant pas d'éliminer les termes d'ordre 2 ou plus

$$\partial_t (U + u^*) + \frac{1}{2} \partial_x (U^2 + 2Uu^* + O(u^{*2})) + \partial_x^3 (U + u^*) = 0$$

or  $U = 0$  donc

$$\partial_t u^* + \partial_x^3 u^* = 0. \quad (1.3)$$

Etant donné que l'on cherche une solution sous la forme d'une onde plane monochromatique (et que l'on travaille en une seule dimension), on pose :

$$u^*(t, x) = e^{i(\xi x - \omega t)} \quad \begin{cases} \xi : \text{vecteur d'onde} \\ \omega : \text{pulsation temporelle} \end{cases}$$

Il ne reste plus qu'à injecter  $u^*$  dans notre équation linéarisée (1.3). On obtient donc

$$-i\omega \cancel{e^{i(\xi x - \omega t)}} + (i\xi)^3 \cancel{e^{i(\xi x - \omega t)}} = 0,$$

c'est-à-dire

$$W(\xi) = \omega \quad \text{avec } W : \xi \mapsto -\xi^3.$$

Vérifions des deux façons possibles si notre EDP est bien dispersive.

$$\begin{aligned} i) \quad & W''(\xi) = -6\xi \neq 0 \\ ii) \quad & \text{Vitesse de groupe : } v_g(\xi) = W'(\xi) = -3\xi^2 \\ & \text{Vitesse de phase : } v_\varphi(\xi) = \frac{W(\xi)}{\xi} = -\xi^2 \end{aligned}$$

On a finalement  $v_g \neq v_\varphi$  qui confirme (i) et donc que notre EDP est dispersive.

### 1.1.2 Conservation de la norme $L^2$

On travaille avec  $u$  de classe  $C^\infty$  et localisée, c'est-à-dire  $\begin{cases} u(t, x = \pm\infty) = 0 & \forall t \in \mathbb{R}_+ \\ \partial_x^k u(t, x = \pm\infty) = 0 & \forall (t, k) \in \mathbb{R}_+ \times \mathbb{N}^* \end{cases}$   
Notons la norme  $L^2$  de (KdV) :

$$\|u\|_{L^2} = \int_{\mathbb{R}} u^2 dx. \quad (\text{L2KdV})$$

En multipliant notre équation de base par  $u$ , elle devient

$$\partial_t uu + u^2 \partial_x u + u \partial_x^3 u = 0. \quad (1.4)$$

Maintenant, il ne reste plus qu'à intégrer (1.4),

$$\int_{\mathbb{R}} \partial_t uu + u^2 \partial_x u + u \partial_x^3 u = 0.$$

On peut simplifier les deux termes par intégrations par parties successives et par localité de  $u$  :

$$\int_{\mathbb{R}} u^2 \partial_x u = \left[ \frac{1}{3} u^3 \right]_{\mathbb{R}} = 0 \quad \text{et} \quad \int_{\mathbb{R}} u \partial_x^3 u = [u \partial_x^2 u]_{\mathbb{R}} - \int_{\mathbb{R}} \partial_x u \partial_x^2 u = - \left[ \frac{1}{2} (\partial_x u)^2 \right]_{\mathbb{R}} = 0.$$

Donc on obtient :

$$\int_{\mathbb{R}} \partial_t (u^2) = 2 \int_{\mathbb{R}} \partial_t uu = 0.$$

Pour justifier la dérivée sous le signe intégrable, nous nous placerons avec  $u$  qui est  $C^\infty$  et à support compact. Nous admettrons la généralisation au cas d'une solution localisée.

- i) Pour tout  $t$ ,  $x \mapsto u^2(t, x)$  intégrable car  $u$  est  $C^\infty$  et à support compact.
- ii) Pour tout  $x$ ,  $t \mapsto u^2(t, x)$  dérivable car  $u$  est supposée  $C^\infty$ .
- iii) On peut dominer  $u^2$  par un majorant de  $u^2$  (indépendant de  $t$  et de  $x$ ) qui est intégrable sur un support compact.

Par le théorème de dérivation sous le signe intégrale et en généralisant au cas d'une solution localisée, nous obtenons finalement la conservation de la norme  $L^2$  de KdV :

$$\frac{d}{dt} \|u\|_{L^2} = \frac{d}{dt} \int_{\mathbb{R}} u^2 dx = 0.$$

### 1.1.3 Conservation de l'Hamiltonien

On travaille toujours avec  $u$  de classe  $C^\infty$  et localisée. Dans notre cas, l'hamiltonien de (KdV) est :

$$H(u) = \int_{\mathbb{R}} \partial_x u^2 - \frac{1}{3} u^3 \, dx. \quad (\text{HKdV})$$

En dérivant notre équation de base (KdV) suivant  $x$ , on obtient une première équation :

$$\partial_x \partial_t u + (\partial_x u)^2 + u \partial_x^2 u + \partial_x^4 u = 0. \quad (1.5)$$

On multiplie ensuite l'équation de KdV par  $u^2$ , on obtient une deuxième équation :

$$\partial_t u u^2 + u^3 \partial_x u + u^2 \partial_x^3 u = 0. \quad (1.6)$$

Maintenant il suffit de multiplier (1.5) par  $2\partial_x u$  tout en soustrayant par (1.6),

$$2\partial_x u \partial_x \partial_t u + 2(\partial_x u)^3 + 2u \partial_x u \partial_x^2 u + 2\partial_x u \partial_x^4 u - \partial_t u u^2 - u^3 \partial_x u - u^2 \partial_x^3 u = 0.$$

En intégrant le tout,

$$\int_{\mathbb{R}} 2\partial_x u \partial_x \partial_t u - \partial_t u u^2 = \int_{\mathbb{R}} \underbrace{-2(\partial_x u)^3}_{(1)} - \underbrace{2u \partial_x u \partial_x^2 u}_{(2)} - \underbrace{2\partial_x u \partial_x^4 u}_{(3)} + \underbrace{u^3 \partial_x u}_{(4)} + \underbrace{u^2 \partial_x^3 u}_{(5)}$$

En intégrant par parties, tout en utilisant la localité de  $u$ , les termes se simplifient :

$$\begin{aligned} \int_{\mathbb{R}} (1) &= \int_{\mathbb{R}} -2(\partial_x u)^3 = -2 [u(\partial_x u)^2]_{\mathbb{R}} + \int_{\mathbb{R}} 4u \partial_x u \partial_x^2 u = \int_{\mathbb{R}} 4u \partial_x u \partial_x^2 u, \\ \int_{\mathbb{R}} (5) &= \int_{\mathbb{R}} u^2 \partial_x^3 u = [\partial_x^2 u u^2]_{\mathbb{R}} - \int_{\mathbb{R}} 2u \partial_x^2 u \partial_x u = - \int_{\mathbb{R}} 2u \partial_x^2 u \partial_x u, \\ \int_{\mathbb{R}} (3) &= [-2\partial_x u \partial_x^3 u]_{\mathbb{R}} + 2 \int_{\mathbb{R}} \partial_x^3 u \partial_x^2 u = [(\partial_x^2 u)^2]_{\mathbb{R}} = 0, \\ \int_{\mathbb{R}} (4) &= \int_{\mathbb{R}} u^3 \partial_x u = \left[ \frac{1}{4} u^4 \right]_{\mathbb{R}} = 0. \end{aligned}$$

Or  $\int_{\mathbb{R}} (1) + (5) + (2) = 0$  donc

$$\begin{aligned} \int_{\mathbb{R}} 2\partial_x u \partial_x \partial_t u - \partial_t u u^2 &= 0, \\ \int_{\mathbb{R}} \partial_t ((\partial_x u)^2 - \frac{1}{3} u^3) &= 0. \end{aligned}$$

Pour justifier la dérivée sous le signe intégrable, nous nous placerons avec  $u$  qui est  $C^\infty$  et à support compact. Nous admettrons la généralisation au cas d'une solution localisée.

- i) Pour tout  $t$ ,  $x \mapsto (\partial_x u(t, x))^2 - \frac{1}{3} u^3(t, x)$  intégrable car  $u$  est  $C^\infty$  à support compact.
- ii) Pour tout  $x$ ,  $t \mapsto (\partial_x u(t, x))^2 - \frac{1}{3} u^3(t, x)$  dérivable car  $u$  est supposée  $C^\infty$ .
- iii) On peut dominer  $(\partial_x u)^2 - \frac{1}{3} u^3$  par un majorant de  $|(\partial_x u)^2 - \frac{1}{3} u^3|$  qui est (indépendant de  $t$  et de  $x$ ) qui est intégrable sur un support compact.

Par le théorème de dérivation sous le signe intégrale et en généralisant au cas d'une solution localisée, nous obtenons finalement la conservation de l'hamiltonien de (KdV) :

$$\frac{d}{dt} H(u) = \frac{d}{dt} \int_{\mathbb{R}} (\partial_x u)^2 - \frac{1}{3} u^3 = 0$$

#### 1.1.4 Invariance d'échelle

L'équation admet une invariance d'échelle [7], c'est-à-dire que si  $u$  est solution alors pour tout réel  $\lambda$ ,  $u_\lambda(t, x) = \lambda^2 u(\lambda^3 t, \lambda x)$  l'est également.

En effet, on a

$$\begin{aligned}\partial_t u_\lambda(t, x) &= \lambda^5 \partial_t u(\lambda^3 t, \lambda x), \\ \partial_x u_\lambda(t, x) &= \lambda^3 \partial_x u(\lambda^3 t, \lambda x), \\ \partial_x^3 u_\lambda(t, x) &= \lambda^5 \partial_x^3 u(\lambda^3 t, \lambda x).\end{aligned}$$

On obtient donc

$$\partial_t u_\lambda + u_\lambda \partial_x u_\lambda + \partial_x^3 u_\lambda = \lambda^5 (\partial_t u + u \partial_x u + \partial_x^3 u)(\lambda^3 t, \lambda x) = \lambda^5 \times 0 = 0 \quad \text{car } u \text{ est solution.}$$

**Explication heuristique du changement d'échelle :** Dans notre cas, notons entre crochet les dimensions des unités et prenons le cas  $\lambda = 0, 1$  avec  $y$  la hauteur de l'eau. On a

$$\begin{aligned}y[m] &= 0,01 \times u(0,001 \times t[s], 0,1 \times x[m]), \\ 100 \times y[m] &= u(t[ms], x[dm]), \\ y[hm] &= u(t[ms], x[dm]).\end{aligned}$$

C'est-à-dire, en choisissant une échelle de temps en  $ms$ , d'espace en  $dm$  et prenons  $hm$  pour l'ordonnée, on obtient exactement la même solution.

#### 1.1.5 Translation espace-temps

Une autre propriété admise par l'équation est que si  $u$  est solution alors  $\forall (t_0, x_0) \in \mathbb{R}_+ \times \mathbb{R}$ ,  $u(t + t_0, x + x_0)$  l'est aussi, appelée translation espace-temps [7]. En effet, on a

$$\begin{aligned}\partial_t (u(t + t_0, x + x_0)) &= \partial_t u(t + t_0, x + x_0), \\ \partial_x (u(t + t_0, x + x_0)) &= \partial_x u(t + t_0, x + x_0), \\ \partial_x^3 (u(t + t_0, x + x_0)) &= \partial_x^3 u(t + t_0, x + x_0).\end{aligned}$$

On obtient donc  $\partial_t (u(t + t_0, x + x_0)) + u(t + t_0, x + x_0) \partial_x (u(t + t_0, x + x_0)) + \partial_x^3 (u(t + t_0, x + x_0)) = (\partial_t u + u \partial_x u + \partial_x^3 u)(t + t_0, x + x_0) = 0$  car  $u$  est solution.

## 1.2 Propriétés vérifiées par le soliton

Dans ce paragraphe, nous nous intéressons aux propriétés vérifiées par le soliton et uniquement le soliton. Un rappel de l'équation de (KdV) ne nous fera pas de mal.

$$\partial_t u(t, x) + u(t, x) \partial_x u(t, x) + \partial_x^3 u(t, x) = 0 \quad \forall (t, x) \in \mathbb{R}_+ \times \mathbb{R}$$

#### 1.2.1 Résolution de l'équation

On veut modéliser une onde progressive (une vague) se propageant à une vitesse  $c > 0$ . Pour cela, on étudie l'équation (KdV) en posant  $u(t, x) = \tilde{u}(y)$  où  $y = x - ct$  [4]. On a, par la règle de la chaîne,

$$\begin{aligned}\partial_t \tilde{u}(t, x) &= -c u'(x - ct) = -c u'(y), \\ \partial_x \tilde{u}(t, x) &= u'(x - ct) = u'(y).\end{aligned}$$

D'après les équations précédentes, (KdV) s'écrit :

$$-c\tilde{u}'(y) + \tilde{u}(y)\tilde{u}'(y) + \tilde{u}^{(3)}(y) = 0. \quad (1.7)$$

En intégrant (1.7) et en multipliant par  $\tilde{u}'(y)$  pour les valeurs de  $y$  telles que  $\tilde{u}'$  soit non nulle, on obtient :

$$-c\tilde{u}(y)\tilde{u}'(y) + \frac{1}{2}\tilde{u}^2(y)\tilde{u}'(y) + \tilde{u}^{(2)}(y)\tilde{u}'(y) = A\tilde{u}'(y).$$

où  $A$  est une constante réelle. De même, en intégrant cette dernière équation, on a :

$$-\frac{c}{2}\tilde{u}^2(y) + \frac{1}{6}\tilde{u}^3(y) + \frac{1}{2}(\tilde{u}'(y))^2 = A\tilde{u}(y) + B \quad (1.8)$$

où  $B$  est aussi une constante réelle. D'après les hypothèses en début de partie, comme notre soliton modélise une vague ou un pulse lumineux, notre solution est localisée. Ainsi en faisant tendre  $y$  vers l'infini dans nos deux équations, on obtient

$$A = 0 \quad \text{et} \quad B = 0.$$

Donc notre équation (1.8) peut s'écrire :

$$-\frac{c}{2}\tilde{u}^2(y) + \frac{1}{6}\tilde{u}^3(y) + \frac{1}{2}(\tilde{u}'(y))^2 = 0.$$

D'après l'équation précédente, on a :

$$(\tilde{u}'(y))^2 = \tilde{u}^2(y)c - \frac{1}{3}\tilde{u}^3(y).$$

$(\tilde{u}')^2$  est positive donc  $\tilde{u}^2c - \frac{1}{3}\tilde{u}^3$  aussi. On en déduit donc :

$$\tilde{u}'(y) = \pm \sqrt{\tilde{u}^2(y)c - \frac{1}{3}\tilde{u}^3(y)} = \pm |\tilde{u}(y)| \sqrt{c - \frac{1}{3}\tilde{u}(y)}.$$

Toujours d'après nos hypothèses sur le soliton, notre solution est strictement positive. Rappelons nous aussi que l'on a supposé que  $\tilde{u}'$  était non nulle donc  $\tilde{u} \neq 3c$ . Nous pouvons appliquer la séparation des variables :

$$\frac{d\tilde{u}}{\tilde{u}\sqrt{c - \frac{1}{3}\tilde{u}}} = \pm dy. \quad (1.9)$$

Posons  $v = \frac{1}{\sqrt{\tilde{u}}}$  alors  $dv = -\frac{d\tilde{u}}{2\tilde{u}\sqrt{\tilde{u}}}$ . Donc on peut exprimer (1.9) en fonction de  $v$  ce qui nous permettra plus tard de reconnaître une dérivée connue,

$$\begin{aligned} \frac{2\cancel{\sqrt{\tilde{u}}}dv}{\cancel{\sqrt{c - \frac{1}{3}\tilde{u}}}} &= \mp dy, \\ \frac{2\cancel{\sqrt{\tilde{u}}}dv}{\sqrt{\frac{\cancel{\sqrt{3}}}{3}}\sqrt{\frac{3c}{\cancel{\tilde{u}}} - 1}} &= \mp dy, \\ 2\sqrt{3}\frac{dv}{\sqrt{3cv^2 - 1}} &= \mp dy. \end{aligned}$$

Or  $\operatorname{argcosh}'(u) = \frac{u'}{\sqrt{u^2-1}}$ . Ainsi on peut réécrire notre dernière équation comme étant :

$$\frac{2}{\sqrt{c}} \operatorname{dargcosh}(\sqrt{3c}v) = \mp dy.$$

En primitivant, on obtient :

$$\frac{2}{\sqrt{c}} \operatorname{argcosh}(\sqrt{3c}v(y)) = \mp y + D$$

où  $D$  est une constante réelle. L'équation étudiée devient donc :

$$v(y) = \frac{1}{\sqrt{3c}} \cosh\left(\frac{\sqrt{c}}{2}(\mp y + D)\right).$$

Or, on a

$$v = \frac{1}{\sqrt{u}},$$

on obtient donc une solution

$$\tilde{u}(y) = \frac{3c}{\cosh^2\left(\frac{\sqrt{c}}{2}(\mp y + D)\right)}$$

que l'on peut finalement écrire en fonction de  $t$  et  $x$ .

$$u(t, x) = \frac{3c}{\cosh^2\left(\frac{\sqrt{c}}{2}(\mp(x - ct) + D)\right)}.$$

Afin de simplifier la suite, on se place de façon à ce que  $\tilde{u}$  possède un maximum en  $y = 0$  et ainsi qu'elle soit symétrique par rapport à l'axe des ordonnées dans le repère en mouvement à la vitesse  $c$ . Pour se faire, cela implique que  $D = 0$ . Comme, de plus, le cosinus hyperbolique est une fonction paire, on obtient :

$$u(t, x) = \frac{3c}{\cosh^2\left(\frac{\sqrt{c}}{2}(x - ct)\right)}. \quad (\text{SKdV})$$

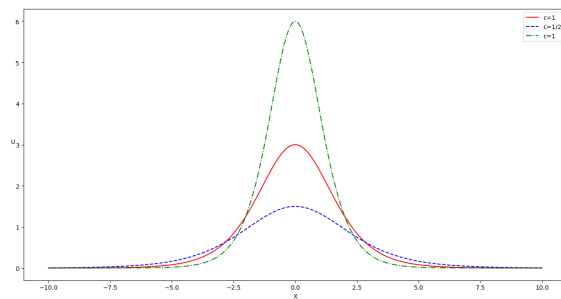


FIGURE 1 – Exemples de soliton à  $t=0$  :  $c=1$  (rouge),  $c=1/2$  (bleu),  $c=2$  (vert)

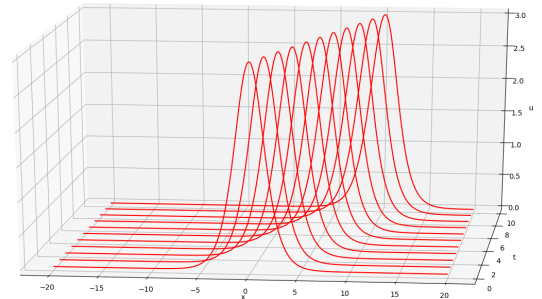


FIGURE 2 – Evolution du soliton en temps et en espace avec  $c=1$



### 1.2.2 Portrait de phase

Pour calculer le portrait de phase de (KdV) on va partir de l'EDO obtenue après l'intégration de (1.7), tout en n'oubliant pas que  $u$  est localisée, c'est-à-dire

$$\tilde{u}^{(2)} + \frac{1}{2}\tilde{u}^2 - c\tilde{u} = 0$$

que nous pouvons réécrire comme ci dessous

$$U' = \begin{pmatrix} \tilde{u}' \\ c\tilde{u} - \frac{1}{2}\tilde{u}^2 \end{pmatrix} \quad \text{avec} \quad U = \begin{pmatrix} \tilde{u} \\ \tilde{u}' \end{pmatrix}. \quad (1.10)$$

On définit l'application  $F$  associée à (1.10) telle que :

$$F : (\tilde{u}, \tilde{u}') \mapsto (\tilde{u}', c\tilde{u} - \frac{1}{2}\tilde{u}^2).$$

On peut apercevoir deux points stationnaires  $\begin{pmatrix} 0 \\ 0 \end{pmatrix}$  et  $\begin{pmatrix} 2c \\ 0 \end{pmatrix}$  que l'on retrouve dans la figure affichée par le code ajouté en annexe.

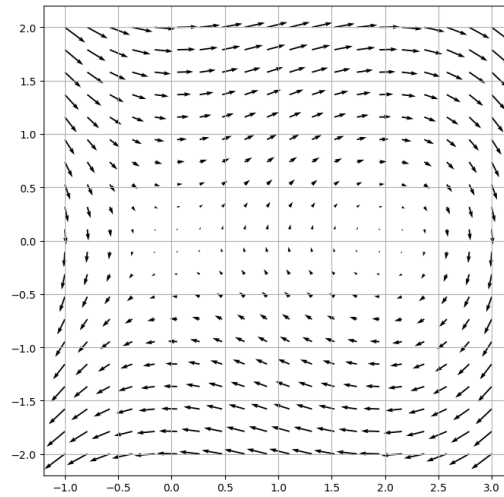


FIGURE 3 – Portrait de Phase de (KdV) pour c=1



## 2 Analyse de l'équation de Schrödinger non linéaire

L'équation de Schrödinger non linéaire (*Nonlinear Schrödinger equation*) communément appelée NLS permet de représenter de nombreux systèmes physiques comme la propagation de la lumière dans les fibres optiques ou encore en optique non linéaire.

$$i\partial_t u(t, x) + \partial_x^2 u(t, x) + |u(t, x)|^{2\sigma} u(t, x) = 0 \quad \forall (t, x, \sigma) \in \mathbb{R}_+ \times \mathbb{R} \times \mathbb{R}_+^* \quad (\text{NLS})$$

$u$  est donc à valeurs complexes et on peut la supposer aussi localisée, c'est-à-dire que le module de la solution et ses dérivées tendent vers 0 quand  $x$  tend vers l'infini.

### 2.1 Propriétés vérifiées par l'équation

Dans ce paragraphe, nous allons étudier les différentes propriétés vérifiées par (NLS). On peut voir facilement que

$$\partial_x \bar{u} = \overline{\partial_x u} \quad \text{et} \quad \partial_t \bar{u} = \overline{\partial_t u}.$$

#### 2.1.1 L'équation de NLS est une EDP dispersive et non linéaire

Tout d'abord, on peut distinguer des termes linéaires  $i\partial_t u$ ,  $\partial_x^2 u$  et des termes non linéaires  $|u|^{2\sigma} u$  dans (NLS). Comme pour (KdV), il faut commencer par linéariser notre équation autour de l'état d'équilibre  $U = 0$  pour étudier sa dispersion :

$$u(t, x) = U + u^*(t, x) \quad (2.1)$$

où  $u^*$  est une petite perturbation. Remplaçons  $u$  dans notre équation (2.1) en n'oubliant pas d'éliminer les termes d'ordre 2 ou plus

$$i\partial_t(U + u^*) + \partial_x^2(U + u^*) + (U + u^*)|U + u^*|^{2\sigma} = 0$$

or  $U = 0$  et  $|U + u^*|^{2\sigma} = O(u^{*2})$ , donc

$$i\partial_t u^* + \partial_x^2 u^* = 0. \quad (2.2)$$

On travaille avec une perturbation  $u^*$  sous la forme d'une onde plane monochromatique, on pose :

$$u^*(t, x) = e^{i(\xi x - \omega t)} \quad \begin{cases} \xi : \text{vecteur d'onde} \\ \omega : \text{pulsation temporelle.} \end{cases}$$

Il ne reste plus qu'à injecter  $u^*$  dans notre équation linéarisée (2.2).

$$W(\xi) = \omega \quad \text{avec} \quad W : \xi \mapsto \xi^2$$

Vérifions des deux façons possibles si notre EDP est bien dispersive

$$i) \quad W''(\xi) = 2 \neq 0$$

$$ii) \quad \text{Vitesse de groupe} : v_g(\xi) = W'(\xi) = 2\xi^2$$

$$\text{Vitesse de phase} : v_\varphi(\xi) = \frac{W(\xi)}{\xi} = \xi$$

On a finalement  $v_g \neq v_\varphi$  qui confirme (i) et donc que notre EDP est dispersive.

### 2.1.2 Conservation de la norme $L^2$

On travaille avec  $u$  de classe  $C^\infty$  et localisée. Notons la norme  $L^2$  de (NLS) :

$$\|u\|_{L^2} = \int_{\mathbb{R}} |u|^2 dx. \quad (\text{L2NLS})$$

Suivons les astuces proposées dans [7]. En multipliant notre équation par  $\bar{u}$ , elle devient :

$$i\partial_t u \bar{u} = -\partial_x^2 u \bar{u} - |u|^{2\sigma+2}.$$

On a

$$2\Re(\partial_t u \bar{u}) = -2\Im(\partial_x^2 u \bar{u}) - 2\Im(|u|^{2\sigma+2}) \quad (2.3)$$

or

$$\int_{\mathbb{R}} \partial_x^2 u \bar{u} = [\partial_x u \bar{u}]_{\mathbb{R}} - \int_{\mathbb{R}} \partial_x u \overline{\partial_x u} = - \int_{\mathbb{R}} |\partial_x u|^2,$$

donc en intégrant (2.3)

$$\int_{\mathbb{R}} 2\Re(\partial_t u \bar{u}) = 0.$$

De plus on a

$$\int_{\mathbb{R}} \partial_t |u|^2 = \int_{\mathbb{R}} \partial_t (u \bar{u}) = \int_{\mathbb{R}} \partial_t u \bar{u} + \partial_t \bar{u} u = \int_{\mathbb{R}} 2\Re(\partial_t u \bar{u}) = 0.$$

Pour justifier la dérivée sous le signe intégrable, nous nous placerons avec  $u$  qui est  $C^\infty$  et à support compact. Nous admettrons la généralisation au cas d'une solution localisée.

- i) Pour tout  $t$ ,  $x \mapsto |u(t, x)|^2$  intégrable car  $u$  est  $C^\infty$  et à support compact.
- ii) Pour tout  $x$ ,  $t \mapsto |u(t, x)|^2$  dérivable car  $u$  est supposée  $C^\infty$ .
- iii) On peut dominer  $|u|^2$  par un majorant de  $|u|^2$  (indépendant de  $t$  et de  $x$ ) qui est intégrable sur un support compact.

Par le théorème de dérivation sous le signe intégrale et en généralisant au cas d'une solution localisée, nous obtenons finalement la conservation de la norme  $L^2$  de (NLS) :

$$\frac{d}{dt} \|u\|_{L^2} = \frac{d}{dt} \int_{\mathbb{R}} |u|^2 dx = 0.$$

### 2.1.3 Conservation du Moment

Notons le moment de (NLS) comme suit [7],

$$M(u) = \Im \left( \int_{\mathbb{R}} \bar{u} \partial_x u dx \right). \quad (\text{MNLS})$$

En multipliant (NLS) par  $-i\overline{\partial_x u}$  tout en intégrant et prenant la partie imaginaire, on obtient une nouvelle équation

$$\underbrace{\Im \left( \int_{\mathbb{R}} \partial_t u \overline{\partial_x u} \right)}_{(1)} - \underbrace{\Im \left( \int_{\mathbb{R}} i \partial_x^2 u \overline{\partial_x u} \right)}_{(2)} - \underbrace{\Im \left( \int_{\mathbb{R}} i |u|^{2\sigma} \overline{\partial_x u} u \right)}_{(3)} = 0$$

avec des termes qui s'annulent :

$$\begin{aligned}
(2) &= i \int_{\mathbb{R}} \partial_x^2 u \overline{\partial_x u} = i [\partial_x u \overline{\partial_x u}]_{\mathbb{R}} - i \int_{\mathbb{R}} \partial_x u \overline{\partial_x^2 u} = i \overline{\int_{\mathbb{R}} \partial_x^2 u \overline{\partial_x u}} = \overline{(2)}, \\
(3) &= i \int_{\mathbb{R}} u |u|^{2\sigma} \overline{\partial_x u} = i [u |u|^{2\sigma} \overline{u}]_{\mathbb{R}} - i \int_{\mathbb{R}} \partial_x u |u|^{2\sigma} \overline{u} - i \int_{\mathbb{R}} \sigma u u^{\sigma-1} \overline{u}^\sigma \partial_x u \overline{u} - i \int_{\mathbb{R}} \sigma u \overline{u}^{\sigma-1} u^\sigma \overline{\partial_x u \overline{u}} \\
&= -i \int_{\mathbb{R}} \partial_x u |u|^{2\sigma} \overline{u} - i \int_{\mathbb{R}} \sigma \overline{u} |u|^{2\sigma} \partial_x u - i \int_{\mathbb{R}} \sigma u |u|^{2\sigma} \overline{\partial_x u} \\
&= -i \int_{\mathbb{R}} (\sigma + 1) \overline{u} |u|^{2\sigma} \partial_x u - i \int_{\mathbb{R}} \sigma u |u|^{2\sigma} \overline{\partial_x u}, \\
(3) + i \int_{\mathbb{R}} \sigma u |u|^{2\sigma} \overline{\partial_x u} &= i \overline{\int_{\mathbb{R}} (\sigma + 1) u |u|^{2\sigma} \overline{\partial_x u}} = (\sigma + 1) \overline{(3)}.
\end{aligned}$$

Or  $(3) = \int_{\mathbb{R}} i |u|^{2\sigma} \overline{\partial_x u} u$ , donc

$$(3) + i \int_{\mathbb{R}} \sigma u |u|^{2\sigma} \overline{\partial_x u} = i \int_{\mathbb{R}} (\sigma + 1) u |u|^{2\sigma} \overline{\partial_x u} = (3)(\sigma + 1).$$

Ainsi  $(2) \in \mathbb{R}$  et  $(3) \in \mathbb{R}$ , on obtient

$$\Im(2) = \Im(3) = 0 \quad \text{et donc} \quad \Im(1) = 0.$$

De plus on a

$$\Im \left( \int_{\mathbb{R}} \partial_t (u \overline{\partial_x u}) \right) = \Im \left( \underbrace{\int_{\mathbb{R}} \partial_t u \overline{\partial_x u}}_{(1)} \right) + \Im \left( \int_{\mathbb{R}} u \overline{\partial_t \partial_x u} \right)$$

or  $\partial_t \overline{\partial_x u} = -i \overline{\partial_x^3 u} - i \overline{\partial_x (|u|^{2\sigma} u)}$ , donc

$$\Im \left( \int_{\mathbb{R}} \partial_t (u \overline{\partial_x u}) \right) = \Im \left( \int_{\mathbb{R}} u \overline{\partial_t \partial_x u} \right) = -\Im \left( \underbrace{\int_{\mathbb{R}} i \overline{\partial_x^3 u} u}_{(4)} \right) - \Im \left( \underbrace{\int_{\mathbb{R}} i \overline{\partial_x (|u|^{2\sigma} u)} u}_{(5)} \right).$$

De plus on réécrit (5) tel que

$$(5) = [i |u|^{2\sigma} \overline{u} u]_{\mathbb{R}} - i \int_{\mathbb{R}} |u|^{2\sigma} \overline{u} \partial_x u.$$

Comme précédemment, par intégrations par parties on obtient  $(4) \in \mathbb{R}$  et  $(5) \in \mathbb{R}$ , ainsi

$$\Im(4) = \Im(5) = 0.$$

Ce qui nous permet d'obtenir finalement :

$$\Im \left( \int_{\mathbb{R}} \partial_t (u \overline{\partial_x u}) \right) = 0.$$

Pour justifier la dérivée sous le signe intégrable, nous nous placerons avec  $u$  qui est  $C^\infty$  et à support compact. Nous admettrons la généralisation au cas d'une solution localisée.

- i) Pour tout  $t$ ,  $x \mapsto u(t, x) \overline{\partial_x u(t, x)}$  intégrable car  $u$  est  $C^\infty$  et à support compact.
- ii) Pour tout  $x$ ,  $t \mapsto u(t, x) \overline{\partial_x u(t, x)}$  dérivable car  $u$  est supposée  $C^\infty$ .
- iii) On peut dominer  $u \overline{\partial_x u}$  par  $|u \overline{\partial_x u}|$  (indépendant de  $t$  et de  $x$ ) qui est intégrable sur un support compact.

Par le théorème de dérivation sous le signe intégrale et en généralisant au cas d'une solution localisée, nous obtenons finalement la conservation du moment de (NLS) :

$$\frac{d}{dt} M(u) = -\frac{d}{dt} \Im \left( \int_{\mathbb{R}} u \overline{\partial_x u} \right) = 0.$$

#### 2.1.4 Conservation de l'Energie

On travaille toujours avec  $u$  de classe  $C^\infty$  et localisée. Dans la culture scientifique, on parle plutôt d'énergie que d'hamiltonien pour (NLS). Dans notre cas, l'énergie de (NLS) s'écrit :

$$E(u) = \int_{\mathbb{R}} \frac{1}{2} |\partial_x u|^2 - \frac{1}{2\sigma + 2} |u|^{2\sigma + 2} dx. \quad (\text{ENLS})$$

En multipliant notre équation de base (NLS) par  $\overline{\partial_t u}$ , on obtient :

$$i |\partial_t u|^2 = -\partial_x^2 u \overline{\partial_t u} - u |u|^{2\sigma} \overline{\partial_t u}.$$

Maintenant il ne reste plus qu'à intégrer et prendre la partie réelle,

$$\Re \left( \int_{\mathbb{R}} i |\partial_t u|^2 \right) = \Re \left( - \int_{\mathbb{R}} \partial_x^2 u \overline{\partial_t u} \right) + \Re \left( - \int_{\mathbb{R}} u |u|^{2\sigma} \overline{\partial_t u} \right).$$

Comme  $i |\partial_t u|^2$  est imaginaire pur, on peut réécrire notre équation comme ci-dessous :

$$0 = \Re \left( - \int_{\mathbb{R}} \partial_x^2 u \overline{\partial_t u} \right) + \Re \left( - \int_{\mathbb{R}} u |u|^{2\sigma} \overline{\partial_t u} \right). \quad (2.4)$$

Tout d'abord, on peut commencer par développer  $\int_{\mathbb{R}} \partial_t |\partial_x u|^2$  en se rappelant que la somme d'un nombre complexe avec son conjugué est égale à deux fois sa partie réelle. De plus,  $u$  est  $C^\infty$  donc on peut appliquer le lemme de Schwartz :

$$\begin{aligned} \int_{\mathbb{R}} \partial_t |\partial_x u|^2 &= \int_{\mathbb{R}} \partial_t (\overline{\partial_x u} \partial_x u) \\ &= \int_{\mathbb{R}} \partial_t \overline{\partial_x u} \partial_x u + \partial_t \partial_x u \overline{\partial_x u} \\ &= \int_{\mathbb{R}} \partial_x \overline{\partial_t u} \partial_x u + \partial_x \partial_t u \overline{\partial_x u} \\ &= [\overline{\partial_t u} \partial_x u + \partial_t u \overline{\partial_x u}]_{\mathbb{R}} - \int_{\mathbb{R}} \overline{\partial_t u} \partial_x^2 u + \partial_t u \overline{\partial_x^2 u} \\ &= -2\Re \left( \int_{\mathbb{R}} \overline{\partial_t u} \partial_x^2 u \right). \end{aligned}$$

Donc on obtient :

$$\Re \left( - \int_{\mathbb{R}} \partial_x^2 u \overline{\partial_t u} \right) = \int_{\mathbb{R}} \partial_t \frac{1}{2} |\partial_x u|^2. \quad (2.5)$$

Ensuite on va développer le deuxième terme de (2.4) .

$$\begin{aligned} 2\Re \left( \int_{\mathbb{R}} u|u|^{2\sigma} \overline{\partial_t u} \right) &= \int_{\mathbb{R}} u|u|^{2\sigma} \overline{\partial_t u} + \overline{u}|u|^{2\sigma} \partial_t u \\ &= \frac{1}{\sigma+1} \int_{\mathbb{R}} \partial_t (u^{\sigma+1} \overline{u}^{\sigma+1}) \\ &= \frac{1}{\sigma+1} \int_{\mathbb{R}} \partial_t (|u|^{2\sigma+2}). \end{aligned}$$

On a donc :

$$\Re \left( \int_{\mathbb{R}} -u|u|^{2\sigma} \overline{\partial_t u} \right) = -\frac{1}{2\sigma+2} \int_{\mathbb{R}} \partial_t (|u|^{2\sigma+2}). \quad (2.6)$$

Grâce à (2.5) et (2.6), on trouve

$$\int_{\mathbb{R}} \frac{1}{2} \partial_t |\partial_x u|^2 - \frac{1}{2\sigma+2} \partial_t (|u|^{2\sigma+2}) = 0.$$

Pour justifier la dérivée sous le signe intégrable, nous nous placerons avec  $u$  qui est  $C^\infty$  et à support compact. Nous admettrons la généralisation au cas d'une solution localisée.

- i) Pour tout  $t$ ,  $x \mapsto \frac{1}{2} |\partial_x u(t, x)|^2 - \frac{1}{2\sigma+2} |u(t, x)|^{2\sigma+2}$  intégrable car  $u$  est  $C^\infty$  à support compact.
- ii) Pour tout  $x$ ,  $t \mapsto \frac{1}{2} |\partial_x u(t, x)|^2 - \frac{1}{2\sigma+2} |u(t, x)|^{2\sigma+2}$  dérivable car  $u$  est supposée  $C^\infty$ .
- iii) On peut dominer  $|\frac{1}{2} |\partial_x u|^2 - \frac{1}{2\sigma+2} |u|^{2\sigma+2}|$  par un majorant (indépendant de  $t$  et de  $x$ ) qui est intégrable sur un support compact.

Par le théorème de dérivation sous le signe intégrale et en généralisant au cas d'une solution localisée, nous obtenons finalement la conservation de l'énergie de (NLS) :

$$\frac{d}{dt} E(u) = \frac{d}{dt} \int_{\mathbb{R}} \frac{1}{2} |\partial_x u|^2 - \frac{1}{2\sigma+2} |u|^{2\sigma+2} dx = 0.$$

### 2.1.5 Invariance d'échelle

L'équation admet une invariance d'échelle [7], c'est-à-dire que si  $u$  est solution alors pour tout réel  $\lambda$ ,  $u_\lambda(t, x) = \lambda^{\frac{1}{\sigma}} u(\lambda^2 t, \lambda x)$  l'est également.

En effet, on a

$$\begin{aligned} \partial_t u_\lambda(t, x) &= \lambda^{\frac{2\sigma+1}{\sigma}} \partial_t u(\lambda^2 t, \lambda x), \\ \partial_x^2 u_\lambda(t, x) &= \lambda^{\frac{2\sigma+1}{\sigma}} \partial_x^2 u(\lambda^2 t, \lambda x). \end{aligned}$$

Donc, on obtient bien la conservation de l'échelle :

$$i\partial_t u_\lambda + \partial_x^2 u_\lambda + u_\lambda |u_\lambda|^{2\sigma} = \lambda^{\frac{2\sigma+1}{\sigma}} (i\partial_t u + \partial_x^2 u + u|u|^{2\sigma})(\lambda^2 t, \lambda x) = 0 \quad \text{car } u \text{ est solution.}$$

### 2.1.6 Invariance de phase

L'équation admet aussi une invariance de phase [7], c'est-à-dire que si  $u$  est solution alors  $u_\theta = e^{i\theta} u$  l'est également pour tout réel  $\theta$ .

En effet, on a

$$\begin{aligned}\partial_t u_\theta &= e^{i\theta} \partial_t u, \\ \partial_x^2 u_\theta &= e^{i\theta} \partial_x^2 u.\end{aligned}$$

Donc on obtient bien la conservation de phase :

$$i\partial_t u_\theta + \partial_x^2 u_\theta + u_\theta |u_\theta|^{2\sigma} = e^{i\theta} (i\partial_t u + \partial_x^2 u + u |u|^{2\sigma})(t, x) = 0 \quad \text{car } u \text{ est solution.}$$

### 2.1.7 Translation espace-temps

L'équation admet elle aussi la translation espace-temps [7] : si  $u$  est solution alors  $\forall (t_0, x_0) \in \mathbb{R}_+ \times \mathbb{R}$ ,  $u(t + t_0, x + x_0)$  l'est aussi. En effet, on a

$$\begin{aligned}\partial_t(u(t + t_0, x + x_0)) &= \partial_t u(t + t_0, x + x_0), \\ \partial_x^2(u(t + t_0, x + x_0)) &= \partial_x^2 u(t + t_0, x + x_0).\end{aligned}$$

Donc on obtient bien la translation espace-temps

$$\begin{aligned}i\partial_t(u(t + t_0, x + x_0)) + \partial_x^2(u(t + t_0, x + x_0)) + u(t + t_0, x + x_0)|u(t + t_0, x + x_0)|^{2\sigma} \\ = (i\partial_t u + \partial_x^2 u + u |u|^{2\sigma})(t + t_0, x + x_0) = 0 \quad \text{car } u \text{ est solution.}\end{aligned}$$

## 2.2 Propriétés vérifiées par le soliton

Dans ce paragraphe, nous nous intéressons aux propriétés vérifiées par le soliton de (NLS). Un rappel de l'équation ne nous fera pas de mal.

$$i\partial_t u(t, x) + \partial_x^2 u(t, x) + |u(t, x)|^{2\sigma} u(t, x) = 0 \quad \forall (t, x, \sigma) \in \mathbb{R}_+ \times \mathbb{R} \times \mathbb{R}_+^*$$

### 2.2.1 Résolution de l'équation

La méthode proposée ici a l'avantage d'être générale et de nous donner la dynamique du soliton et non seulement son enveloppe. En annexe se trouve une méthode plus rapide mais moins générale et qui nous donne l'enveloppe du soliton seulement. (NLS) étant complexe, on cherche  $u(t, x) = \phi(t, x)e^{i\theta(t, x)}$  avec  $\phi$  et  $\theta$  des fonctions réelles,  $\phi$  étant positive et localisée en espace [13]. Supposons que l'on traite le cas où  $u$  est différente de la fonction nulle (donc  $\phi \neq 0$ ). En injectant  $u$  dans l'équation, on obtient :

$$(i\partial_t \phi - \partial_t \theta \phi + \partial_x^2 \phi + i\partial_x \theta \partial_x \phi + i\partial_x^2 \theta \phi + i\partial_x \theta \partial_x \phi - (\partial_x \theta)^2 \phi + \phi^{2\sigma+1}) e^{i\theta} = 0.$$

En séparant les parties réelles et imaginaires, on obtient le système suivant :

$$\begin{aligned}-\partial_t \theta \phi + \partial_x^2 \phi - (\partial_x \theta)^2 \phi + \phi^{2\sigma+1} &= 0 \\ \partial_t \phi + 2\partial_x \theta \partial_x \phi + \partial_x^2 \theta \phi &= 0.\end{aligned}$$

On cherche des solutions particulières telles que  $\theta$  se propage à la vitesse  $v$  et  $\phi$  se propage à la vitesse  $c$ . Ainsi, on peut poser  $\phi(t, x) = \tilde{\phi}(x - ct)$  et  $\theta(t, x) = \tilde{\theta}(x - vt)$ . De plus, notons  $y = x - ct$  et  $z = x - vt$ . Par règle de la chaîne, on obtient :

$$\begin{aligned}\partial_x \phi &= \tilde{\phi}', \\ \partial_t \phi &= -c\tilde{\phi}', \\ \partial_x \theta &= \tilde{\theta}', \\ \partial_t \theta &= -v\tilde{\theta}'.\end{aligned}$$



Nous avons donc les relations suivantes :  $\partial_t \phi = -c \partial_x \phi$  et  $\partial_t \theta = -v \partial_x \theta$ . Donc notre système équivaut à :

$$v \partial_x \theta \phi + \partial_x^2 \phi - (\partial_x \theta)^2 \phi + \phi^{2\sigma+1} = 0 \quad (2.7)$$

$$-c \partial_x \phi + 2 \partial_x \theta \partial_x \phi + \partial_x^2 \theta \phi = 0. \quad (2.8)$$

La deuxième équation (2.8) s'intègre (par rapport à  $x$ ) sans trop de difficulté après multiplication par  $\phi$  et devient :

$$-\frac{c}{2} \phi^2 + \partial_x \theta \phi^2 = A_t$$

où  $A_t$  est une constante réelle dépendant de  $t$ . Ici nous cherchons des solutions localisées spatialement en excluant la solution identiquement nulle (qui conduirait à  $\phi = 0$ ). On note que la constante est donc nulle (par exemple en faisant tendre  $x$  vers plus ou moins l'infini) et l'on obtient alors

$$\partial_x \theta = \tilde{\theta}' = c/2.$$

Par une simple intégration (par rapport à  $z$ ), on obtient

$$\tilde{\theta} = \frac{c}{2}(x - vt) + B$$

où  $B$  est une constante réelle. En choisissant l'origine des temps, on peut imposer  $B = 0$ . Maintenant, en reportant cette expression dans la première expression de notre système (2.7), on obtient

$$\frac{cv}{2} \phi + \partial_x^2 \phi - \frac{c^2}{4} \phi + \phi^{2\sigma+1} = 0. \quad (2.9)$$

Notre équation s'intègre facilement après multiplication par  $\partial_x \phi$  (que nous supposons non nul) et devient

$$\frac{cv}{4} (\phi)^2 + \frac{1}{2} (\partial_x \phi)^2 - \frac{c^2}{8} \phi^2 + \frac{1}{2\sigma+2} \phi^{2\sigma+2} = C_t$$

avec  $C_t$  indépendant de  $x$  mais dépendant de  $t$ . N'oublions pas que  $\phi$  est localisée, alors on obtient

$$(\partial_x \phi)^2 + \frac{2cv - c^2}{4} (\phi)^2 + \frac{1}{\sigma+1} \phi^{2\sigma+2} = 0.$$

Or  $(\partial_x \phi)^2 \geq 0$  car  $\phi$  est réelle. Donc  $\frac{2cv - c^2}{4} (\phi)^2 + \frac{1}{\sigma+1} \phi^{2\sigma+2} \leq 0$  quelle que soit  $\phi$ . Comme on a supposé que  $\phi$  est non nulle, on peut donc simplifier par  $\phi^2$ . Ensuite, si l'on fait tendre  $x$  vers  $\pm\infty$ ,  $\phi$  tend vers 0 et on obtient  $\frac{2cv - c^2}{4} \leq 0$  donc  $2cv - c^2 \leq 0$ . Supposons  $c \neq 0$ , si  $c > 0$ ,  $v \leq \frac{c}{2}$  sinon  $c < 0$  et  $v \geq \frac{c}{2}$ . On en conclut donc que  $v$  et  $c$  sont différentes. Notons  $a = \frac{c^2 - 2cv}{4}$ .

$$(\partial_x \phi)^2 - a(\phi)^2 + \frac{1}{\sigma+1} \phi^{2\sigma+2} = 0 \quad (2.10)$$

Comme  $\partial_x \phi = \tilde{\phi}'$ ,  $\phi = \tilde{\phi}$  et que de plus  $a\tilde{\phi}^2 - \frac{1}{\sigma+1} \tilde{\phi}^{2\sigma+1} \geq 0$  car  $(\tilde{\phi}')^2 \geq 0$ , l'équation (2.10) devient :

$$\begin{aligned} \tilde{\phi}' &= \pm \sqrt{a\tilde{\phi}^2 - \frac{1}{\sigma+1} \tilde{\phi}^{2\sigma+2}}, \\ \tilde{\phi}' &= \pm |\tilde{\phi}| \sqrt{a - \frac{1}{\sigma+1} \tilde{\phi}^{2\sigma}}. \end{aligned}$$

Nous étudions la solution  $u$  non nulle avec  $\phi$  positive, ainsi  $\tilde{\phi} \neq 0$ . Rappelons-nous aussi que l'on a supposé que  $\partial_x \phi$  était non nulle donc  $\tilde{\phi}^{2\sigma} \neq a(\sigma+1)$ . Nous pouvons appliquer la séparation des variables :

$$\frac{d\tilde{\phi}}{\tilde{\phi}\sqrt{a - \frac{1}{\sigma+1}\tilde{\phi}^{2\sigma}}} = \pm dy \quad (2.11)$$

Posons  $\psi = \frac{1}{\sqrt{\tilde{\phi}}}$  alors  $d\psi = -\frac{d\tilde{\phi}}{2\tilde{\phi}\sqrt{\tilde{\phi}}}$  Donc (2.11) devient :

$$\begin{aligned} -\frac{2d\psi\sqrt{\tilde{\phi}}}{\sqrt{a - \frac{1}{\sigma+1}\tilde{\phi}^{2\sigma}}} &= \pm dy, \\ \frac{2d\psi\sqrt{\tilde{\phi}}}{\frac{\tilde{\phi}^\sigma}{\sqrt{\sigma+1}}\sqrt{\frac{a(\sigma+1)}{\tilde{\phi}^{2\sigma}} - 1}} &= \mp dy. \end{aligned}$$

Or  $\psi^\sigma = \frac{1}{\tilde{\phi}^{\frac{\sigma}{2}}}$  ce qui nous permet d'écrire l'équation seulement en fonction de  $\psi$  et de faire apparaître la dérivée d'une fonction connue :

$$\frac{d\psi\psi^{2\sigma-1}}{\sqrt{a(\sigma+1)\psi^{4\sigma} - 1}} 2\sqrt{\sigma+1} = \mp dy. \quad (2.12)$$

Comme on l'a déjà vu avec (KdV),  $\operatorname{argcosh}'(u) = \frac{u'}{\sqrt{u^2-1}}$  donc  $\operatorname{argcosh}'(u^{2\sigma}) = \frac{2\sigma u^{2\sigma-1}u'}{\sqrt{u^{4\sigma}-1}}$ . Ainsi on peut réécrire (2.12) :

$$\frac{2\sqrt{\sigma+1} d\operatorname{argcosh}(\sqrt{a(\sigma+1)}\psi^{2\sigma})}{2\sigma\sqrt{a(\sigma+1)}} = \mp dy.$$

En primitivant la dernière équation (par rapport à  $y$ ), on obtient :

$$\frac{1}{\sigma\sqrt{a}} \operatorname{argcosh}(\sqrt{a(\sigma+1)}\psi^{2\sigma}) = \mp y + D$$

où  $D$  est une constante réelle. Ainsi on a :

$$\psi(y) = \left( \frac{\cosh(\sigma\sqrt{a}(\mp y + D))}{\sqrt{a(\sigma+1)}} \right)^{\frac{1}{2\sigma}},$$

or  $\psi = \frac{1}{\sqrt{\tilde{\phi}}}$ , donc on a finalement

$$\tilde{\phi}(y) = \left( \frac{a(\sigma+1)}{\cosh^2(\sigma\sqrt{a}(\mp y + D))} \right)^{\frac{1}{2\sigma}} = \left( \frac{\frac{c^2-2cv}{4}(\sigma+1)}{\cosh^2\left(\sigma\sqrt{\frac{c^2-2cv}{4}}(\mp y + D)\right)} \right)^{\frac{1}{2\sigma}}.$$

Afin de simplifier la suite, on se place de façon à ce que  $\tilde{\phi}$  possède un maximum en  $y = 0$  et ainsi qu'elle soit symétrique par rapport à l'axe des ordonnées. Pour se faire, il faut que  $D = 0$ .

Comme, de plus, le cosinus hyperbolique est une fonction paire, on obtient :

$$u(t, x) = \left( \frac{\frac{c^2 - 2cv}{4}(\sigma + 1)}{\cosh^2 \left( \sigma \sqrt{\frac{c^2 - 2cv}{4}}(x - ct) \right)} \right)^{\frac{1}{2\sigma}} e^{i \frac{\sigma}{2}(x - vt)}. \quad (\text{SNLS})$$

On retrouve bien le soliton ainsi que sa dynamique proposée dans [8].

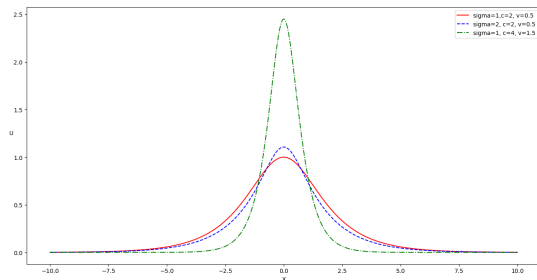


FIGURE 4 – Exemples de soliton à  $t=0$  :  $\sigma = 1$ ,  $c = 2$ ,  $v = 0.5$  (rouge) ;  $\sigma = 2$ ,  $c = 2$ ,  $v = 0.5$  (bleu) ;  $\sigma = 1$ ,  $c = 4$ ,  $v = 1.5$  (vert)

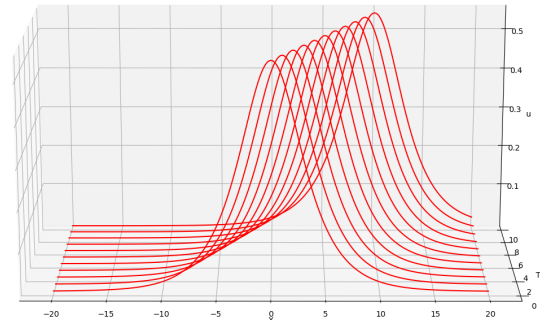


FIGURE 5 – Evolution du soliton en temps et en espace avec  $\sigma = 1$ ,  $c = 2$  et  $v = 0.5$

### 2.2.2 Portrait de phase

Pour calculer le portrait de phase de (NLS), on va partir de l'EDO (2.9) exprimée par  $\tilde{\phi}$

$$\tilde{\phi}'' + \frac{2cv - c^2}{4} \tilde{\phi} + \tilde{\phi}^{2\sigma+1} = 0$$

que nous pouvons réécrire comme ci dessous :

$$\Phi' = \begin{pmatrix} \tilde{\phi}' \\ \frac{c^2 - 2cv}{4} \tilde{\phi} - \tilde{\phi}^{2\sigma+1} \end{pmatrix} \quad \text{avec} \quad \Phi = \begin{pmatrix} \tilde{\phi} \\ \tilde{\phi}' \end{pmatrix}. \quad (2.13)$$

On définit l'application  $F$  associée à (2.13) tel que :

$$F : (\tilde{\phi}, \tilde{\phi}') \mapsto (\tilde{\phi}', \frac{c^2 - 2cv}{4} \tilde{\phi} - \tilde{\phi}^{2\sigma+1})$$

Prenons par exemple  $\sigma = 1$ . On peut apercevoir deux points stationnaires  $\begin{pmatrix} 0 \\ 0 \end{pmatrix}$ ,  $\begin{pmatrix} \sqrt{\frac{c^2 - 2cv}{4}} \\ 0 \end{pmatrix}$  et  $\begin{pmatrix} -\sqrt{\frac{c^2 - 2cv}{4}} \\ 0 \end{pmatrix}$  que l'on retrouve dans la figure affichée par le code ajouté en annexe.

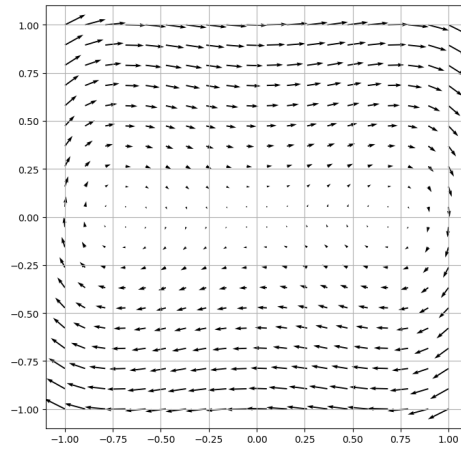


FIGURE 6 – Portrait de phase de NLS pour  $\sigma = 1$ ,  $c = 2$  et  $v = 0.5$

### 3 Résolution numérique des deux équations

Il est intéressant d'avoir des schémas numériques qui se rapprochent des propriétés physiques des EDP. Dans notre cas, les propriétés physiques étudiées sont l'existence des solitons, la conservation de la norme  $L^2$ , la conservation du moment ou encore la conservation de l'hamiltonien. En effet, certains schémas numériques ne sont pas aptes à capter les solitons et certaines de leurs propriétés, d'où la nécessité de schémas spécifiques. Nous allons donc étudier deux types de schémas : un schéma compact symplectique [11] et la méthode de tir [9].

#### 3.1 Résolution de l'équation de KdV par un schéma compact symplectique

Le schéma proposé par Kanazawa *et al.* [11] est un schéma compact, nous verrons plus tard pourquoi on peut dire qu'il est compact, et symplectique, c'est-à-dire qu'il conserve certaines propriétés physiques.

On fixe un entier  $J > 0$  correspondant au nombre de mailles en espace. On définit  $J+1$  points spaciaux qui seront les noeuds du maillage :  $x_j$  pour  $j \in \llbracket 0, J \rrbracket$ . On suppose pour simplifier qu'ils sont équidistants de même pas d'espace avec  $L$  la longueur du maillage étudié. Nous avons donc :

$$x_j = j\Delta x \quad \forall j \in \llbracket 0, J \rrbracket \quad \text{avec } \Delta x = L/J.$$

Donnons-nous une fonction  $u$  (pour l'instant ne dépendant que de l'espace) dont nous allons approximer  $u(x_j)$  par  $v_j$ . De plus supposons que notre fonction est périodique et donc que notre approximation assure la condition suivante :  $v_j = v_{j \pm J}$ .

##### 3.1.1 L'opérateur des différences finies compact

Prenons les mêmes notations que dans l'article [11] et notons  $\delta_c^{(1)}$  l'opérateur des différences finies compact qui discrétise  $\partial_x$ . Il est défini tel que

$$\begin{aligned} & \left( \delta_c^{(1)} v \right)_j + \alpha \left( \left( \delta_c^{(1)} v \right)_{j+1} + \left( \delta_c^{(1)} v \right)_{j-1} \right) + \beta \left( \left( \delta_c^{(1)} v \right)_{j+2} + \left( \delta_c^{(1)} v \right)_{j-2} \right) \\ &= a \frac{v_{j+1} - v_{j-1}}{2\Delta x} + b \frac{v_{j+2} - v_{j-2}}{4\Delta x} + c \frac{v_{j+3} - v_{j-3}}{6\Delta x} \end{aligned}$$

que nous pouvons réécrire matriciellement comme suit

$$A \delta_c^{(1)} v = Bv$$

avec  $A = \begin{pmatrix} 1 & \alpha & \beta & & & & \\ & \alpha & \beta & & & & \\ & \alpha & \beta & & & & \\ & \beta & \alpha & & & & \\ & & \beta & \alpha & & & \\ & & & \beta & \alpha & & \\ & & & & \beta & \alpha & 1 \end{pmatrix}$  et  $B = \begin{pmatrix} 0 & a/2\Delta x & b/4\Delta x & c/6\Delta x & & & -c/6\Delta x & -b/4\Delta x & -a/2\Delta x \\ -a/2\Delta x & & & & & & -c/6\Delta x & -b/4\Delta x & \\ -b/4\Delta x & & & & & & & -c/6\Delta x & \\ -c/6\Delta x & & & & & & & & \\ & c/6\Delta x & b/4\Delta x & a/2\Delta x & & & & & \\ & b/4\Delta x & a/2\Delta x & & & & & & \\ a/2\Delta x & b/4\Delta x & c/6\Delta x & & & & -c/6\Delta x & -b/4\Delta x & -a/2\Delta x & 0 \end{pmatrix}$

où  $\alpha$ ,  $\beta$ ,  $a$ ,  $b$  et  $c$  sont des constantes réelles. On peut noter que lorsque l'on prend ce jeu de variable

$$\alpha = \beta = 0, \quad a = \frac{3}{2}, \quad b = -\frac{3}{5}, \quad c = \frac{1}{10} \quad (C1)$$

on retrouve l'opérateur central des différences finies d'ordre six. Dans l'article, le meilleur jeu de valeur trouvé est le suivant, qui est aussi d'ordre six,

$$\begin{aligned} \alpha &= 0.5381301488732363, & \beta &= 0.066633190123881123, & a &= 1.367577724399269, \\ b &= 0.8234281701082790, & c &= 0.018520783486686603 \end{aligned} \quad (\text{C2})$$

dont on verra l'efficacité plus tard. On parle de schéma compact car, comme précisé dans [11], on obtient une précision d'ordre 6 alors que l'on utilise un stencil beaucoup moins grand (6 autres points) que si on utilisait uniquement les différences finies où il faudrait 8 autres points.

Maintenant travaillons avec l'opérateur  $\delta_c^{(1)}$  dans le cas général. L'opérateur est antisymétrique ce qui nous permettra plus tard d'avoir la conservation de l'hamiltonien. Pour cela considérons deux fonctions  $f$  et  $g$  dont on approximera  $f(x_j)$  et  $g(x_j)$  par  $f_j$  et  $g_j$ .

**Lemme 1** (Propriété d'antisymétrie).

$$\sum_{j=0}^J f_j \left( \delta_c^{(1)} g \right)_j \Delta x = - \sum_{j=0}^J \left( \delta_c^{(1)} f \right)_j g_j \Delta x. \quad (\text{AS})$$

*Démonstration.* On peut voir facilement que  $A$  est symétrique ( $A^T = A$ ) et  $B$  antisymétrique ( $B^T = -B$ ). De plus avec les conditions (C1) et (C2), notre matrice  $A$  est inversible. On a  $A$  circulante dans  $A^{-1}$  aussi. De plus  $A^{-1}$  et  $B$  sont commutatives car se sont deux matrices circulantes. On a donc  $(A^{-1}B)^T f = B^T A^{-T} = -BA^{-1} = -A^{-1}B$ . Nous avons finalement

$$\begin{aligned} \sum_{j=0}^J f_j \left( \delta_c^{(1)} g \right)_j \Delta x &= \sum_{j=0}^J f_j (A^{-1}Bg)_j \Delta x \\ &= \sum_{j=0}^J \left( (A^{-1}B)^T f \right)_j g_j \Delta x \\ &= - \sum_{j=0}^J (A^{-1}Bf)_j g_j \Delta x \\ &= - \sum_{j=0}^J \left( \delta_c^{(1)} f \right)_j g_j \Delta x. \end{aligned}$$

□

### 3.1.2 Conservation de l'hamiltonien discret

Considérons notre équation

$$\partial_t u(t, x) + u(t, x) \partial_x u(t, x) + \partial_x^3 u(t, x) = 0 \quad \forall (t, x) \in \mathbb{R}_+ \times \mathbb{R}. \quad (\text{KdV})$$

On a  $\frac{1}{2}H(u) = \int_{\mathbb{R}} -\frac{1}{6}u^3 + \frac{1}{2}(\partial_x u)^2 dx$  que l'on peut écrire comme  $\frac{1}{2}H(u) = \int_{\mathbb{R}} G(u) dx$  avec  $G$  la densité de l'hamiltonien :

$$G(u) = -\frac{1}{6}u^3 + \frac{1}{2}(\partial_x u)^2. \quad (3.1)$$

Ensuite cherchons l'expression de la petite variation de  $G$  suivant  $u$ ,

$$\begin{aligned} H(u+v) &= H(u) + dH(u).v + o(\|v\|), \\ \int_{\mathbb{R}} G(u+v) &= \int_{\mathbb{R}} G(u) + \left\langle \frac{\delta G}{\delta u}(u) | v \right\rangle_{L^2} + o(\|v\|). \end{aligned}$$

En faisant passer  $\int_{\mathbb{R}} G(u)$  à gauche et en faisant rentrer les termes d'ordre 2 ou plus dans le petit  $o$  :

$$\begin{aligned} \int_{\mathbb{R}} -\frac{1}{6}(u+v)^3 + \frac{1}{6}u^3 + \frac{1}{2}(\partial_x u + \partial_x v)^2 - \frac{1}{2}(\partial_x u)^2 &= \left\langle \frac{\delta G}{\delta u}(u) | v \right\rangle_{L^2} + o(\|v\|), \\ \int_{\mathbb{R}} -\frac{1}{6}(v^3 + 3uv^2 + 3u^2v) + \frac{1}{2}((\partial_x v)^2 + 2\partial_x u \partial_x v) &= \int_{\mathbb{R}} \frac{\delta G}{\delta u}(u)v + o(\|v\|), \\ \int_{\mathbb{R}} -\frac{1}{2}u^2v + \partial_x u \partial_x v &= \int_{\mathbb{R}} \frac{\delta G}{\delta u}(u)v + o(\|v\|). \end{aligned}$$

Par une intégration par parties du deuxième terme du membre de gauche, en utilisant le fait que  $u$  soit une solution localisée, on obtient l'équation suivante

$$\int_{\mathbb{R}} -\frac{1}{2}u^2v - \partial_x^2 uv = \int_{\mathbb{R}} \frac{\delta G}{\delta u}(u)v + o(\|v\|)$$

dont on peut déduire facilement la petite variation de  $G$  suivant  $u$  :

$$\frac{\delta G}{\delta u}(u) = -\frac{1}{2}u^2 - \partial_x^2 u. \quad (3.2)$$

Réécrivons (KdV) à l'aide de notre dernière équation (3.2),

$$\partial_t u = \partial_x \left( -\frac{1}{2}u^2 - \partial_x^2 u \right) = \partial_x \frac{\delta G}{\delta u}(u).$$

On se fixe maintenant un nombre maximal d'itérations temporelles  $N > 0$  ainsi que des temps intermédiaires  $t^n = n\Delta t$  tous séparés du même pas de temps  $\Delta t$  défini par  $\Delta t = \frac{T}{N}$  où  $T$  est la durée d'étude de notre fonction. Nous avons donc :

$$t^n = n\Delta t \quad \forall n \in \llbracket 0, N \rrbracket \quad \text{avec } \Delta t = \frac{T}{N}.$$

On cherche à calculer une valeur approchée de la solution  $u$  en chaque noeud  $(t^n, x_j)$ . On notera  $v_j^n$  cette solution approchée de  $u(t^n, x_j)$  et  $V^n = (v_0^n, \dots, v_j^n)^t$ . Prenons le même schéma proposé par Kanazawa *et al.* dans [11] :

$$\frac{v_j^{n+1} - v_j^n}{\Delta t} = \left( \delta_c^{(1)} \frac{\delta G}{\delta(V^{n+1}, V^n)} \right)_j, \quad (3.3)$$

$$G(V^n)_j = -\frac{1}{6}(v_j^n)^3 + \frac{1}{2} \left( \delta_c^{(1)} V^n \right)_j^2, \quad (3.4)$$

$$\frac{\delta G}{\delta(V^{n+1}, V^n)}_j = -\frac{(v_j^{n+1})^2 + v_j^{n+1}v_j^n + (v_j^n)^2}{6} - \left( \left( \delta_c^{(1)} \right)^2 \frac{(V^{n+1} + V^n)}{2} \right)_j. \quad (3.5)$$

Il ne reste plus qu'à prouver que l'hamiltonien discret est lui aussi conservé avec nos deux discrétisations précédentes. Nous définissons l'hamiltonien discret par  $H(v^n) = \sum_{j=0}^J G(v_j^n) \Delta x$ . Tout d'abord on a

$$\sum_{j=0}^J [G(v_j^{n+1}) - G(v_j^n)] \frac{\Delta x}{\Delta t} = \sum_{j=0}^J \left[ -\frac{1}{6}(v_j^{n+1})^3 + \frac{1}{6}(v_j^n)^3 + \frac{1}{2} \left( \delta_c^{(1)} V^{n+1} \right)_j^2 - \frac{1}{2} \left( \delta_c^{(1)} V^n \right)_j^2 \right] \frac{\Delta x}{\Delta t}. \quad (3.6)$$

L'opérateur satisfait, de plus, le lemme suivant qui nous sera bien utile plus tard.

**Lemme 2.**

$$\sum_{j=0}^J \left( \left( \delta_c^{(1)} \right)^2 V^{n+1} \right)_j v_j^{n+1} - \left( \left( \delta_c^{(1)} \right)^2 V^n \right)_j v_j^n = \sum_{j=0}^J \left( \left( \delta_c^{(1)} \right)^2 (V^{n+1} + V^n) \right)_j (v_j^{n+1} - v_j^n) \quad (\text{L})$$

*Démonstration.*

$$\begin{aligned} \sum_{j=0}^J \left( \left( \delta_c^{(1)} \right)^2 (V^{n+1} + V^n) \right)_j (v_j^{n+1} - v_j^n) &= \sum_{j=0}^J \left( \left( \delta_c^{(1)} \right)^2 V^{n+1} \right)_j v_j^{n+1} - \left( \left( \delta_c^{(1)} \right)^2 V^n \right)_j v_j^n \\ &\quad + \left( \left( \delta_c^{(1)} \right)^2 V^n \right)_j v_j^{n+1} - \left( \left( \delta_c^{(1)} \right)^2 V^{n+1} \right)_j v_j^n. \end{aligned}$$

D'après la propriété d'antisymétrie (AS), on a deux termes qui s'annulent :

$$\begin{aligned} \sum_{j=0}^J \left( \left( \delta_c^{(1)} \right)^2 V^n \right)_j v_j^{n+1} - \left( \left( \delta_c^{(1)} \right)^2 V^{n+1} \right)_j v_j^n &= \sum_{j=0}^J - \left( \delta_c^{(1)} V^{n+1} \right)_j \left( \delta_c^{(1)} V^n \right)_j + \left( \delta_c^{(1)} V^{n+1} \right)_j \left( \delta_c^{(1)} V^n \right)_j \\ &= 0 \end{aligned}$$

Nous obtenons bien ce que l'on voulait.  $\square$

En utilisant notre lemme (L) et la propriété (AS), on peut réécrire l'équation suivante :

$$\begin{aligned} \sum_{j=0}^J \left[ \left( \delta_c^{(1)} V^{n+1} \right)_j^2 - \left( \delta_c^{(1)} V^n \right)_j^2 \right] &= \sum_{j=0}^J \left[ \left( \delta_c^{(1)} V^{n+1} \right)_j \left( \delta_c^{(1)} V^{n+1} \right)_j - \left( \delta_c^{(1)} V^n \right)_j \left( \delta_c^{(1)} V^n \right)_j \right] \\ &= - \sum_{j=0}^J \left[ \left( \left( \delta_c^{(1)} \right)^2 V^{n+1} \right)_j v_j^{n+1} - \left( \left( \delta_c^{(1)} \right)^2 V^n \right)_j v_j^n \right] \\ &= - \sum_{j=0}^J \left[ \left( \left( \delta_c^{(1)} \right)^2 (V^{n+1} + V^n) \right)_j (v_j^{n+1} - v_j^n) \right]. \end{aligned}$$

Reportons cette équation dans (3.6) tout en utilisant l'identité géométrique et notre schéma (3.3).

$$\begin{aligned} \sum_{j=0}^J [G(v_j^{n+1}) - G(v_j^n)] \frac{\Delta x}{\Delta t} &= \sum_{j=0}^J \left[ \left( -\frac{(v_j^{n+1})^2 + v_j^{n+1}v_j^n + (v_j^n)^2}{6} - \frac{1}{2} \left( \left( \delta_c^{(1)} \right)^2 (V^{n+1} + V^n) \right)_j \right) (v_j^{n+1} - v_j^n) \right] \frac{\Delta x}{\Delta t} \\ &= \sum_{j=0}^J \left[ \frac{\delta G}{\delta(V^{n+1}, V^n)_j} \frac{v_j^{n+1} - v_j^n}{\Delta t} \right] \Delta x \\ &= \sum_{j=0}^J \left[ \frac{\delta G}{\delta(V^{n+1}, V^n)_j} \left( \delta_c^{(1)} \frac{\delta G}{\delta(V^{n+1}, V^n)} \right)_j \right] \Delta x. \end{aligned}$$

Il ne reste plus qu'à appliquer notre propriété d'antisymétrie et le fait que deux termes opposés égaux équivalent au terme nul.

$$\begin{aligned} \sum_{j=0}^J \left[ \frac{\delta G}{\delta(V^{n+1}, V^n)_j} \left( \delta_c^{(1)} \frac{\delta G}{\delta(V^{n+1}, V^n)} \right)_j \right] \Delta x &= - \sum_{j=0}^J \left[ \left( \delta_c^{(1)} \frac{\delta G}{\delta(V^{n+1}, V^n)} \right)_j \frac{\delta G}{\delta(V^{n+1}, V^n)_j} \right] \Delta x \\ &= 0 \end{aligned}$$



Ainsi nous avons obtenu la conservation de l'hamiltonien discret :

$$(H(v^{n+1}) - H(v^n)) \Delta x = \sum_{j=0}^J [G(v_j^{n+1}) - G(v_j^n)] \frac{\Delta x}{\Delta t} = 0$$

### 3.1.3 Implémentation du schéma

Afin de simplifier les notations, nous supposons dans ce paragraphe que toutes les opérations entre vecteurs se font terme à terme (puissance, produit, etc). D'après (3.3) et (3.5) nous avons le schéma discret suivant :

$$\frac{v_j^{n+1} - v_j^n}{\Delta t} = \left( -\delta_c^{(1)} \frac{(V^{n+1})^2 + V^{n+1}V^n + (V^n)^2}{6} \right)_j - \left( \left( \delta_c^{(1)} \right)^3 \frac{(V^{n+1} + V^n)}{2} \right)_j.$$

Réécrivons cette équation avec uniquement les termes  $V^{n+1}$  et  $V^n$

$$\left( V^{n+1} + \frac{\Delta t}{6} \delta_c^{(1)} \left( (V^{n+1})^2 + V^{n+1}V^n \right) + \frac{\Delta t}{2} \left( \delta_c^{(1)} \right)^3 V^{n+1} - V^n + \frac{\Delta t}{6} \delta_c^{(1)} (V^n)^2 + \frac{\Delta t}{2} \left( \delta_c^{(1)} \right)^3 V^n \right)_j = 0.$$

Afin de déterminer le vecteur  $V^{n+1}$  en fonction du vecteur  $V^n$ , il convient d'utiliser la méthode de Newton-Raphson décrite dans les rappels. Pour cela définissons la fonction  $F_n$  comme suit où  $x$  est un vecteur de taille  $J+1$

$$F_n(x) = x - V^n + \frac{\Delta t}{6} A^{-1} B \left( x^2 + xV^n + (V^n)^2 \right) + \frac{\Delta t}{2} (A^{-1} B)^3 (x + V^n).$$

Pour appliquer la méthode de Newton-Raphson, il ne reste plus qu'à déterminer la Jacobienne de  $F_n$ . Prenons un scalaire  $h$  et un vecteur  $y$  de norme unitaire

$$\begin{aligned} F_n(x + hy) &= x + hy - V^n + \frac{\Delta t}{6} A^{-1} B \left( (x + hy)^2 + (x + hy)V^n + (V^n)^2 \right) + \frac{\Delta t}{2} (A^{-1} B)^3 (x + hy + V^n) \\ &= x + hy - V^n + \frac{\Delta t}{6} A^{-1} B \left( x^2 + 2xhy + (x + hy)V^n + (V^n)^2 \right) + \frac{\Delta t}{2} (A^{-1} B)^3 (x + hy + V^n) + o(h) \\ &= F_n(x) + hy + \frac{\Delta t}{6} A^{-1} B (2xhy + hyV^n) + \frac{\Delta t}{2} (A^{-1} B)^3 hy + o(h) \\ &= F_n(x) + \left( I_{J+1} + \frac{\Delta t}{6} A^{-1} B I_{2x+V^n} + \frac{\Delta t}{2} (A^{-1} B)^3 \right) hy + o(h) \end{aligned}$$

où  $I_{2x+V^n}$  correspond à une matrice diagonale de taille  $J+1$  dont  $(I_{2x+V^n})_{i,i} = (2x + V^n)_i$  et  $I_{J+1}$  est la matrice identité de taille  $J+1$ . Nous obtenons donc la Jacobienne de  $F_n$  que l'on notera  $J_n$  :

$$J_n(x) = I_{J+1} + \frac{\Delta t}{6} A^{-1} B I_{2x+V^n} + \frac{\Delta t}{2} (A^{-1} B)^3.$$

Finalement à l'aide du code mis en annexe, nous pouvons commencer par tracer en figure 7 l'évolution du soliton de l'équation de KdV avec le jeu de variable (C2) et les variables :  $L = 30$ ,  $J = 200$ ,  $T = 40$  et  $N = 2000$ . De plus, nous pouvons voir en figure 8 que la condition (C1) est beaucoup moins optimale que (C2), pour cela on se place à  $t = 14$ . Les conditions aux bords étant périodiques, les calculs ont été effectués sur le vecteur  $(V^n)_{j < J}$ . Toutes les matrices implémentées dans le code (y compris la matrice jacobienne de  $F_n$ ) sont donc de taille  $J$  avec les conditions aux bords incluses dans les coefficients en haut à droite et en bas à gauche.

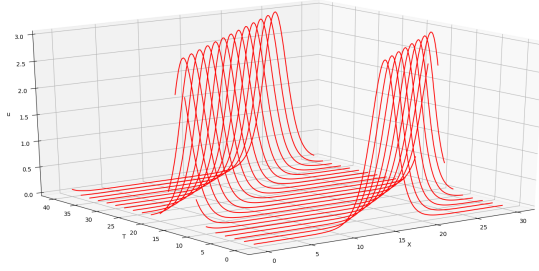


FIGURE 7 – Evolution du soliton avec  $L = 30$ ,  $J = 200$ ,  $T = 40$  et  $N = 2000$

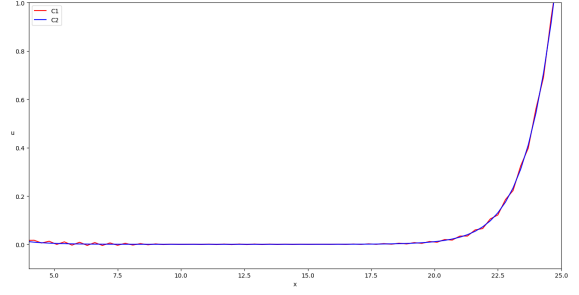


FIGURE 8 – Comparaison du soliton à  $t = 14$  suivant (C1) ou (C2)

On peut voir également en figure 9 que le jeu (C1) ne conserve pas l'hamiltonien comparé à l'autre jeu de paramètre (C2).

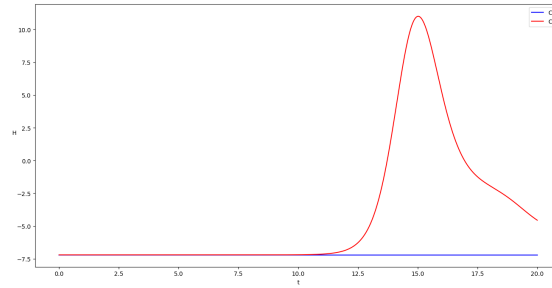


FIGURE 9 – Evolution de l'hamiltonien en fonction du temps suivant (C1) ou (C2)

Nous pouvons tenter d'utiliser la méthode de Heun (jointe dans les rappels) afin de résoudre notre équation

$$\frac{dV}{dt} = -V \left( \delta_c^{(1)} V \right) - \delta_c^{(1)} \delta_c^{(2)} V$$

où  $V(t) = (v_0(t), \dots, v_J(t))$  et  $\delta_c^{(2)}$  correspond à l'opérateur discret de  $\partial_x^2$ . Dans notre cas, on se placera à la condition (C1) qui nous permettra d'écrire  $\delta_c^{(2)} = \left( \delta_c^{(1)} \right)^2$  ce qui nous simplifiera les calculs. On peut donc réécrire notre équation telle que

$$\frac{dV}{dt} = -V \left( \delta_c^{(1)} V \right) - \left( \delta_c^{(1)} \right)^3 V.$$

Notons  $X$  notre discrétisation spatiale, c'est-à-dire  $X = (x_0, \dots, x_j, \dots, x_J)$ . Il ne nous reste plus qu'à appliquer la méthode de Heun avec :

$$V^{n+1} = V^n + \frac{\Delta t}{2} (k_1 + k_2)$$

$$\text{où } \begin{cases} V^0 = u(X, 0), \\ G(t, V^n) = -V^n (A^{-1} B V^n) - (A^{-1} B)^3 V^n, \\ k_1 = G(t + n\Delta t, V^n), \\ k_2 = G(t + (n+1)\Delta t, V^n + \Delta t k_1). \end{cases}$$

Finalement, grâce aux figures 10 et 11 ci-dessous, on peut conclure qu'à contrario le schéma de Heun ne capte ni le soliton de l'équation de KdV, ni la conservation de l'hamiltonien.

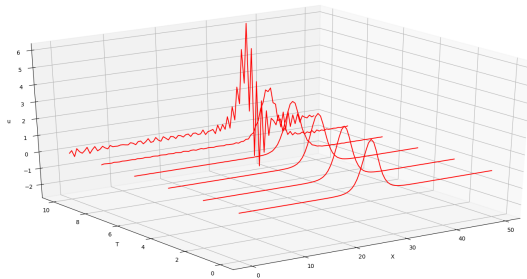


FIGURE 10 – Evolution du soliton du schéma de Heun avec  $L = 30$ ,  $J = 100$ ,  $T = 10$  et  $N = 500$

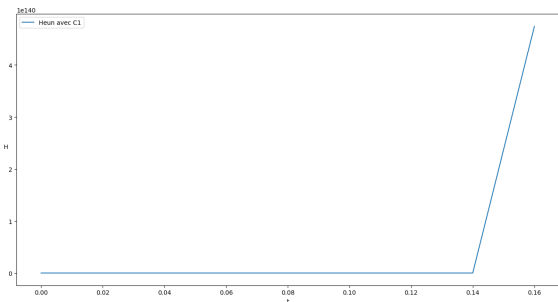


FIGURE 11 – Evolution de l'hamiltonien du schéma de Heun en fonction du temps avec les conditions (C1)

## 3.2 Résolution de l'équation de NLS par la méthode de tir

La méthode de tir est une méthode qui permet de capter l'enveloppe de la solution recherchée mais pas sa dynamique. L'avantage de cette méthode est qu'elle permet de trouver la solution recherchée sans avoir à connaître toutes les conditions initiales.

### 3.2.1 Introduction

Prenons l'exemple d'un simple lancer de balle, lancé depuis une hauteur de 10 mètres ( $x = 0, y = 10$ ) avec une certaine vitesse initiale donnée par  $v_0 = (v_{0x}, v_{0y})$  où  $v_{0x} = v_0 \cos(\alpha)$  et  $v_{0y} = v_0 \sin(\alpha)$ .  $v_0$  est la vitesse initiale de l'objet et  $\alpha = 20^\circ$  est l'angle de lancement. Dans notre problème, nous voulons modéliser la trajectoire de la balle pour qu'elle atterrisse à une distance de 15 mètres. Il s'agit donc d'un problème aux limites (sans connaître la condition initiale) or seuls les problèmes de Cauchy (avec condition initiale connue et valeurs aux limites inconnues) sont facilement résolubles. C'est là que devient intéressante la méthode de tir. Elle remplace un problème aux limites par un problème de Cauchy dont les conditions initiales sont calculées par dichotomie. En effet, il nous suffit de modéliser le tracé pour une valeur  $v_0^1$  telle que la balle atterrisse après 15 mètres. Il ne reste plus qu'à appliquer une dichotomie afin d'approcher la vitesse initiale qui permettra à la balle d'atterrir à 15 mètres.

Remarque : Dans le cas présent du lancé de balle, l'expression exacte de la hauteur de chute est connue grâce à la deuxième loi de Newton :  $y(x) = -\frac{gx^2}{2v_{0x}^2} + \frac{v_{0y}x}{v_{0x}} + y_0$ . Mais dans la plupart des cas, l'expression exacte reste inconnue et la méthode de tir est très intéressante.

Il nous suffit ensuite de modéliser le tracé pour une valeur  $v_0^1$  telle que la balle atterrisse avant 15 mètres et pour une autre valeur  $v_0^2$  telle que la balle atterrisse après 15 mètres. Il ne reste plus qu'à appliquer une dichotomie afin d'approcher la vitesse initiale qui permettra à la balle d'atterrir à 15 mètres.

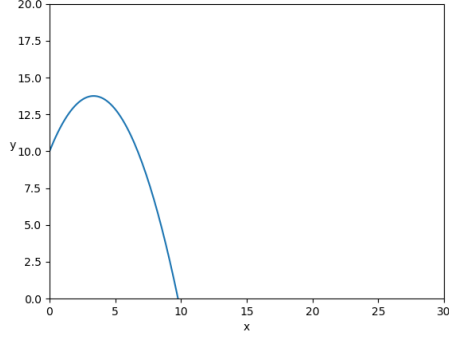


FIGURE 12 – Trajectoire de la balle pour  $v_0 = 3$

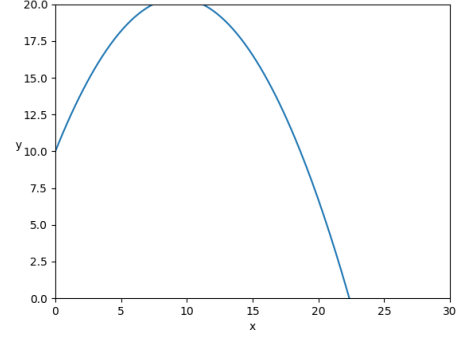


FIGURE 13 – Trajectoire de la balle pour  $v_0 = 5$

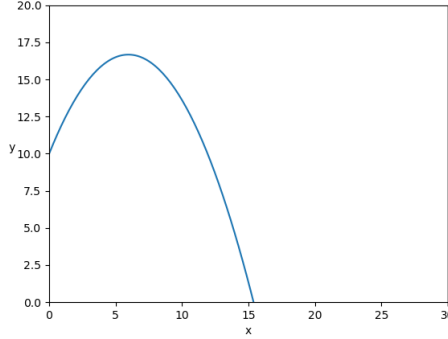


FIGURE 14 – Trajectoire de la balle pour  $v_0 = \frac{3+5}{2} = 4$

### 3.2.2 Au tour de l'équation de NLS

Maintenant, nous allons appliquer le même principe mais à (NLS). Pour cela on se ramène à une EDO d'ordre 2. Prenons la solution  $u(t, x) = \phi(x)e^{i\omega t}$ , trouvée par la méthode utilisée en annexe, qui n'est rien d'autre que l'enveloppe du soliton.

$$-\omega\phi + \phi'' + |\phi|^{2\sigma}\phi = 0$$

Comme mentionné dans l'article de Di Menza [9], on peut se ramener au cas où  $\omega = 1$  afin d'éliminer une variable dans notre schéma. En effet, si  $u$  est solution alors, par invariance d'échelle, pour tout réel  $\lambda$ ,  $u_\lambda(t, x) = \lambda^{1/\sigma}u(\lambda^2 t, \lambda x) = \tilde{\phi}(x)e^{i\tilde{\omega}t}$  l'est également où  $\tilde{\phi}(x) = \lambda^{1/\sigma}\phi(\lambda x)$  et  $\tilde{\omega} = \omega\lambda^2$ . Injectons  $u_\lambda$  dans (NLS) :

$$-\tilde{\omega}\tilde{\phi} + \tilde{\phi}'' + |\tilde{\phi}|^{2\sigma}\tilde{\phi} = 0.$$

Prenons  $\lambda = 1/\sqrt{\omega}$ , c'est-à-dire  $\tilde{\omega} = 1$ , ce qui nous donne une équation correctement "redimensionnée" :

$$-\tilde{\phi} + \tilde{\phi}'' + |\tilde{\phi}|^{2\sigma}\tilde{\phi} = 0.$$

Posons  $X(x) = \begin{pmatrix} \tilde{\phi}(x) \\ \tilde{\phi}'(x) \end{pmatrix}$  et  $F(x, X(x)) = \begin{pmatrix} \tilde{\phi}'(x) \\ (1 - |\tilde{\phi}(x)|^{2\sigma}) \tilde{\phi}(x) \end{pmatrix}$ . Alors nous obtenons le système suivant :

$$\begin{cases} X'(x) = F(x, X(x)) \\ X(0) = \begin{pmatrix} \alpha \\ 0 \end{pmatrix} \end{cases}$$

où  $\alpha = \tilde{\phi}(0)$  est l'inconnue à trouver à l'aide de la méthode de tir. On sait que  $\tilde{\phi}'(0) = 0$  car on cherche à modéliser un soliton dont le maximum se situe à l'origine.

On fixe un entier  $J > 0$  correspondant au nombre de mailles en espace. On définit  $J+1$  points spaciaux qui seront les noeuds du maillage :  $x_j$  pour  $j \in \llbracket 0, J \rrbracket$ . On suppose pour simplifier qu'ils sont équidistants de même pas d'espace avec  $L$  la longueur du maillage étudié. Nous avons donc :

$$x_j = j\Delta x \quad \forall j \in \llbracket 0, J \rrbracket \quad \text{avec } \Delta x = L/J.$$

On cherche à calculer une valeur approchée de la solution  $\tilde{\phi}$  en chaque noeud  $x_j$ . On notera  $\tilde{\phi}_j$  cette solution approchée de  $\tilde{\phi}(x_j)$ . De plus, on notera  $X_j$  le vecteur approché de  $X(x_j)$ . Pour résoudre cette EDO, nous utiliserons le schéma de RK4 (rappelé au début) tel que :

$$X_{j+1} = X_j + \frac{\Delta x}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

$$\text{où } \begin{cases} k_1 = F(x_j, X_j), \\ k_2 = F\left(x_j + \frac{\Delta x}{2}, X_j + \frac{\Delta x}{2}k_1\right), \\ k_3 = F\left(x_j + \frac{\Delta x}{2}, X_j + \frac{\Delta x}{2}k_2\right), \\ k_4 = F(x_j + \Delta x, X_j + \Delta x k_3). \end{cases}$$

Par symétrie du soliton par rapport à l'axe des ordonnées, nous ne travaillons que dans le domaine des  $x$  positifs. On admettra qu'il existe de plus une (unique) solution strictement positive qui tend vers 0 à l'infini [9]. Ces deux critères nous seront utiles pour notre méthode de tir. Prenons un intervalle  $[a_0, b_0]$  tel que si  $\alpha = b_0$  alors il existe au moins une valeur négative ou nulle et si  $\alpha = a_0$  alors, à partir d'un certain  $x_j$ , il existe au moins une valeur supérieure ou égale à un  $\varepsilon$ . Comme décrit dans [9], si pour  $\alpha_0 = (a_0 + b_0)/2$  on obtient une courbe possédant au moins une valeur négative alors on pose  $[a_1 = a_0, b_1 = \alpha_0]$  sinon on obtiendra une courbe possédant, à partir d'un certain  $x_j$ , au moins une valeur supérieure ou égale à un certain  $\varepsilon$  et alors là on posera  $[a_1 = \alpha_0, b_1 = b_0]$ . Au rang  $n$  de notre dichotomie, on obtiendra l'intervalle  $[a_n, b_n]$  le critère d'arrêt étant la satisfaction des deux conditions citées précédemment. Dans notre cas, on prendra  $\varepsilon = 10^{-8}$ . La méthode n'étant pas assez précise pour prendre un  $\varepsilon$  plus petit au risque que le code ne se termine pas. On choisira de plus  $a_0 = 0.9$  et  $b_0 = 2$ , ce qui nous permettra d'obtenir les figures 15, 16 et 17 suivantes.

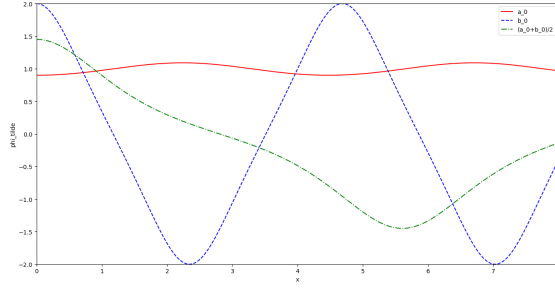


FIGURE 15 – Evolution de  $\tilde{\phi}$  en fonction de  $x$  pour les 3 premières itérations ( $a_0 = 0.9$ ,  $b_0 = 3$ )

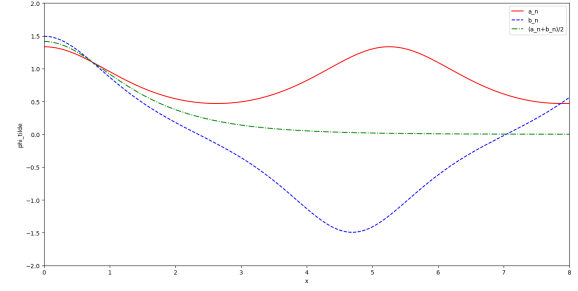


FIGURE 16 – Evolution de  $\tilde{\phi}$  en fonction de  $x$  pour 3 itérations au rang  $n$  ( $a_n = 1,3342135$ ,  $b_n = 1,4542135$ )

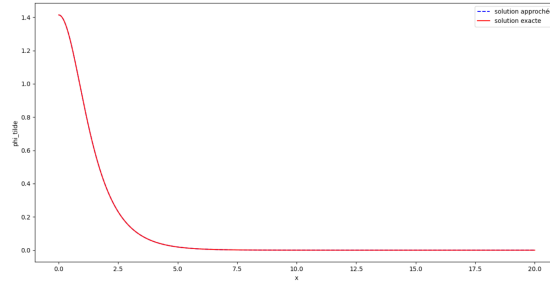


FIGURE 17 – Evolution de la solution approchée et exacte suivant  $x$

Comme nous pouvons l'apercevoir dans la dernière figure (17), la méthode de tir nous permet bien de capter le soliton de (NLS). Néanmoins permet-elle de conserver l'énergie? Si c'est le cas, pour chaque solution au problème de Cauchy de condition initiale  $(\alpha_0)$  on obtiendrait la même énergie (car l'énergie est conservée, c'est-à-dire constant, et c'est une propriété propre à l'équation et non au soliton). Cependant, comme nous le confirme la figure d'après, la méthode de tir ne permet pas la conservation de l'énergie.

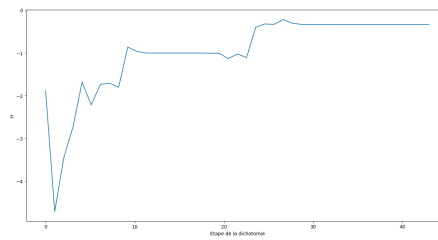


FIGURE 18 – Evolution de l'énergie de l'équation de NLS suivant les différentes itérations du code

### 3.3 Adaptation des schémas à l'autre équation

Dans ce paragraphe, il est question de savoir s'il est possible d'adapter le schéma compact de [11] à (NLS) et la méthode de tir à (KdV).

#### 3.3.1 Méthode de tir pour l'équation de KdV

Commençons par adapter la méthode de tir à (KdV) qui semble être l'adaptation la moins compliquée à réaliser. Basons-nous sur le même principe que [9] mais avec l'EDO d'ordre 2 obtenue après intégration de (1.7), c'est-à-dire :

$$\tilde{u}^{(2)}(y) - c\tilde{u}(y) + \frac{1}{2}\tilde{u}^2(y) = 0.$$

Nous rappelons ici que la solution  $\tilde{u}$  est localisée en espace, aucune constante d'intégration n'intervient donc. Posons  $U(y) = \begin{pmatrix} \tilde{u}(y) \\ \tilde{u}'(y) \end{pmatrix}$  et  $F(y, U(y)) = \begin{pmatrix} \tilde{u}'(y) \\ (c - \frac{1}{2}\tilde{u}(y))\tilde{u}(y) \end{pmatrix}$ . Alors nous obtenons le système suivant :

$$\begin{cases} U'(y) = F(y, U(y)) \\ U(0) = \begin{pmatrix} \alpha \\ 0 \end{pmatrix} \end{cases}$$

où  $\alpha = \tilde{u}(0)$  est l'inconnue à trouver à l'aide de la méthode de tir. On sait que  $\tilde{u}'(0) = 0$  car on cherche à modéliser un soliton dont le maximum se situe à l'origine.

On fixe un entier  $J > 0$  correspondant au nombre de mailles en espace. On définit  $J+1$  points spaciaux qui seront les noeuds du maillage :  $y_j$  pour  $j \in \llbracket 0, J \rrbracket$ . On suppose pour simplifier qu'ils sont équidistants de même pas d'espace avec  $L$  la longueur du maillage étudié. Nous avons donc :

$$y_j = j\Delta y \quad \forall j \in \llbracket 0, J \rrbracket \quad \text{avec } \Delta y = L/J.$$

On cherche à calculer une valeur approchée de la solution  $\tilde{u}$  en chaque noeud  $y_j$ . On notera  $\tilde{u}_j$  cette solution approchée de  $\tilde{u}(y_j)$ . De plus, on notera  $Y_j$  le vecteur approché de  $Y(y_j)$ . Pour résoudre cette EDO, nous utiliserons le schéma de RK4 tel que :

$$Y_{j+1} = Y_j + \frac{\Delta y}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

où  $\begin{cases} k_1 = F(y_j, U_j), \\ k_2 = F\left(y_j + \frac{\Delta y}{2}, U_j + \frac{\Delta y}{2}k_1\right), \\ k_3 = F\left(y_j + \frac{\Delta y}{2}, U_j + \frac{\Delta y}{2}k_2\right), \\ k_4 = F(y_j + \Delta y, U_j + \Delta y k_3), \end{cases}$

On admettra qu'il existe de plus une (unique) solution strictement positive qui tend vers 0 à l'infini. Ces deux critères nous seront utiles pour notre adaptation de la méthode de tir. Par tâtonnement, on peut voir que si on prend un  $\alpha$  trop petit alors on modélise une solution qui possède des valeurs supérieures à un certain  $\varepsilon$  à partir d'un certain  $y_j$ . Si notre  $\alpha$  est trop grand, la solution prend au moins une valeur négative. Nous sommes donc dans la même situation que pour (NLS), prenons un intervalle  $[a_0, b_0]$  tel que si  $\alpha = b_0$  alors il existe au moins une valeur négative ou nulle et si  $\alpha = a_0$  alors, à partir d'un certain  $x_j$ , il existe au moins une valeur supérieure ou égale à un  $\varepsilon$ .

En prenant les mêmes variables que notre première méthode de tir, c'est-à-dire  $L = 20$ ,  $J = 2000$  et  $\varepsilon = 10^{-8}$ , l'algorithme ne se terminait pas. Il a donc fallu trouver l'origine du

problème. A première vue, il semblerait que le problème vienne d'un  $\varepsilon$  trop petit. Cependant même avec  $\varepsilon = 10^{-1}$ , le code ne s'arrêtait pas. En testant la méthode de tir avec  $\alpha = 3$ , qui est la valeur que l'on doit trouver pour notre soliton avec  $c = 1$ , nous obtenons le même problème : un code qui ne se termine pas. En modélisant à part la solution trouvée pour  $\alpha = 3$  et  $L = 40$ , on obtient la figure suivante :

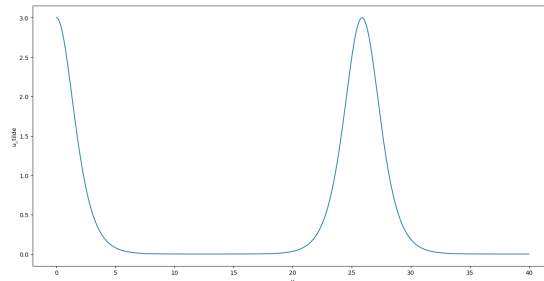


FIGURE 19 – Problème de périodicité pour la méthode de tir appliquée à l'équation de KdV

Or notre application  $F$  est  $C^\infty$  donc, d'après le théorème de Cauchy-Lipschitz, il devrait exister une unique solution au problème de Cauchy de solution initiale  $\alpha = 3$ . On peut donc déduire que cette périodicité n'est pas un problème physique mais numérique. Cependant, nous utilisons exactement le même code que pour (NLS). On peut donc déduire que l'origine du problème devrait venir d'une accumulation de petites erreurs qui deviennent assez grandes à partir d'un moment. En effet, en résolvant l'EDO avec la fonction "odeint" de python, pour  $L = 25$ ,  $J = 20000$  et  $\alpha = 3$ , on obtient la courbe rouge de la figure 20. Alors qu'avec le schéma RK4, on trouve la courbe bleue (de la figure 20). On peut voir que pour la même EDO avec les mêmes paramètres, la courbe bleue semble mener à une solution périodique alors que la courbe rouge à une solution anti-périodique. Ceci est impossible d'après le théorème de Cauchy-Lipschitz : il y a unicité de la solution.

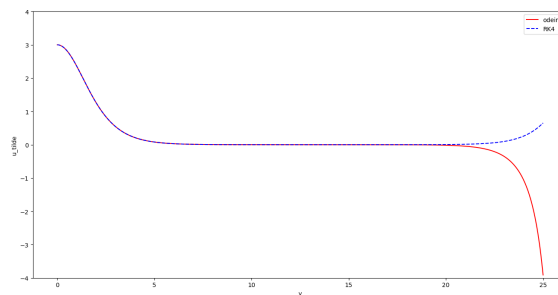


FIGURE 20 – Comparaison de la solution de l'EDO par deux méthodes différentes

Pour ne plus avoir ces effets purement numériques, il suffit de raffiner la grille de calcul servant à intégrer l'EDO. En utilisant la fonction "odeint" de python, il suffit de rajouter  $rtol = 1.e - 13$ ,  $atol = 1.e - 13$  qui sont respectivement la tolérance relative et la tolérance absolue. Par défaut, ces tolérances sont de  $1.49012 \times 10^{-8}$ . Cela est insuffisant dans notre cas. En utilisant le schéma RK4, il suffit de raffiner  $J$ . Par exemple  $J = 10000$  pour  $L = 25$  convient parfaitement.



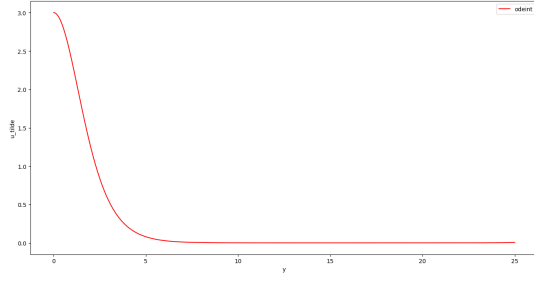


FIGURE 21 – Résolution avec odeint pour  $L = 25$  et  $rtol = atol = 10^{-13}$

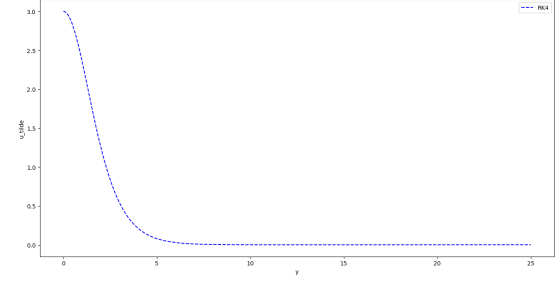


FIGURE 22 – Résolution avec RK4 pour  $L = 25$  et  $J = 10000$

Si l'on choisit d'augmenter la longueur de  $L$ , il faudra raffiner encore plus  $J$  (pour RK4) ou diminuer encore plus la tolérance numérique (pour "odeint"). Prenons  $L = 30$ . En utilisant "odeint" il faudrait une tolérance numérique plus faible que  $10^{-13}$  ce qui n'a plus de sens pour un ordinateur. Par contre, RK4 reste encore tout à fait valable. Si l'on prend  $J = 40000$ , on obtient le soliton de (KdV). Cependant, le temps de calcul est beaucoup plus long.

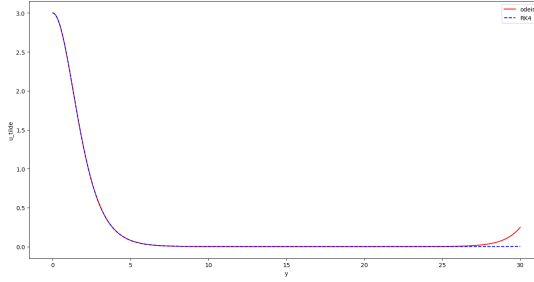


FIGURE 23 – Comparaison entre "odeint" pour une tolérance de  $10^{-13}$  et RK4 pour  $J = 40000$  avec  $L = 30$

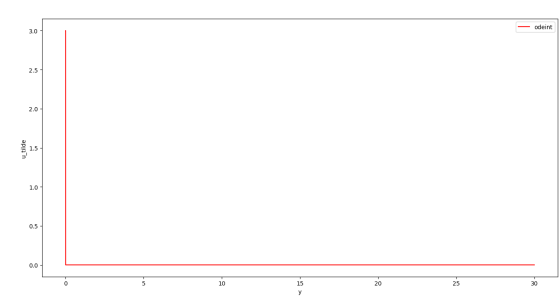


FIGURE 24 – Résolution avec odeint pour  $L = 30$  et  $rtol = atol = 10^{-14}$

### 3.3.2 Schéma compact pour l'équation de NLS

L'adaptation du schéma compact à (NLS) va demander plus de travail que la méthode de tir appliquée à (KdV). En effet, il va falloir construire notre propre discrétisation du schéma tout en conservant son énergie. Nous nous demanderons également en fin de paragraphe s'il est possible de conserver de plus la norme  $L^2$  et le moment. Commençons par nous rappeler l'équation.

$$i\partial_t u(t, x) + \partial_x^2 u(t, x) + |u(t, x)|^{2\sigma} u(t, x) = 0 \quad \forall (t, x, \sigma) \in \mathbb{R}_+ \times \mathbb{R} \times \mathbb{R}_+^*$$

#### 3.3.2.1 Calcul de la variation de la densité de l'énergie de NLS

La solution  $u$  est à valeurs complexes donc on peut écrire comme suit  $u(t, x) = a(t, x) + ib(t, x)$  avec  $a$  et  $b$  deux fonctions  $C^\infty$  à valeurs réelles. Supposons que l'on écarte la solution nulle qui n'est pas intéressante. Ainsi (NLS) se réécrit :

$$i\partial_t a - \partial_t b + \partial_x^2 a + i\partial_x^2 b + (a^2 + b^2)^\sigma (a + ib) = 0.$$

En séparant la partie réelle et la partie imaginaire, on obtient le système suivant :

$$\begin{aligned}\partial_t a &= -\partial_x^2 b - (a^2 + b^2)^\sigma b \\ \partial_t b &= \partial_x^2 a + (a^2 + b^2)^\sigma a.\end{aligned}\tag{3.7}$$

De même nous pouvons réécrire l'énergie (ENLS) en fonction de  $a$  et  $b$

$$E(a, b) = \int_{\mathbb{R}} \frac{1}{2} (\partial_x a^2 + \partial_x b^2) - \frac{1}{2\sigma + 2} (a^2 + b^2)^{\sigma+1}$$

où l'on note  $\tilde{E}(a, b) = \frac{1}{2} (\partial_x a^2 + \partial_x b^2) - \frac{1}{2\sigma+2} (a^2 + b^2)^{\sigma+1}$  (DENLS) la densité de l'énergie.

Rappelons-nous que pour le schéma compact de (KdV), il avait fallu trouver la variation de la densité de l'hamiltonien. En se basant sur l'article [11], cherchons les variations  $\frac{\delta \tilde{E}}{\delta a}$  et  $\frac{\delta \tilde{E}}{\delta b}$  afin de réécrire notre système comme nous avons réécrit (KdV) en fonction de  $\frac{\delta G}{\delta u}$  :

$$\begin{aligned}E(a + h\alpha, b) &= \int_{\mathbb{R}} \frac{1}{2} [(\partial_x a + h\partial_x \alpha)^2 + (\partial_x b)^2] - \frac{1}{2\sigma + 2} [(a + h\alpha)^2 + b^2]^{\sigma+1} \\ E(a, b + h\beta) &= \int_{\mathbb{R}} \frac{1}{2} [(\partial_x b + h\partial_x \beta)^2 + (\partial_x a)^2] - \frac{1}{2\sigma + 2} [(b + h\beta)^2 + a^2]^{\sigma+1}.\end{aligned}$$

En négligeant les termes d'ordre 2, on obtient

$$\begin{aligned}E(a + h\alpha, b) &= \int_{\mathbb{R}} \frac{1}{2} [(\partial_x a)^2 + 2h\partial_x \alpha \partial_x a + (\partial_x b)^2] - \frac{1}{2\sigma + 2} [a^2 + 2ha\alpha + b^2]^{\sigma+1} + o(h) \\ E(a, b + h\beta) &= \int_{\mathbb{R}} \frac{1}{2} [(\partial_x a)^2 + 2h\partial_x \beta \partial_x b + (\partial_x b)^2] - \frac{1}{2\sigma + 2} [a^2 + 2hb\beta + b^2]^{\sigma+1} + o(h).\end{aligned}$$

que l'on peut réécrire (car  $a^2 + b^2 \neq 0$  sinon  $|u| = 0$  ce qui est impossible vu que l'on n'étudie pas la solution nulle) :

$$\begin{aligned}E(a + h\alpha, b) &= \int_{\mathbb{R}} \frac{1}{2} [(\partial_x a)^2 + 2h\partial_x \alpha \partial_x a + (\partial_x b)^2] - \frac{1}{2\sigma + 2} (a^2 + b^2)^{\sigma+1} \left[1 + \frac{2ha\alpha}{a^2 + b^2}\right]^{\sigma+1} + o(h) \\ E(a, b + h\beta) &= \int_{\mathbb{R}} \frac{1}{2} [(\partial_x a)^2 + 2h\partial_x \beta \partial_x b + (\partial_x b)^2] - \frac{1}{2\sigma + 2} (a^2 + b^2)^{\sigma+1} \left[1 + \frac{2hb\beta}{a^2 + b^2}\right]^{\sigma+1} + o(h).\end{aligned}$$

Nous pouvons appliquer la formule du DL de  $(1+x)^\alpha$  à  $\left(1 + \frac{2ha\alpha}{a^2+b^2}\right)^{\sigma+1}$  et  $\left(1 + \frac{2hb\beta}{a^2+b^2}\right)^{\sigma+1}$  avec  $h$  qui tend vers 0. Ainsi on obtient le système suivant :

$$\begin{aligned}E(a + h\alpha, b) &= \int_{\mathbb{R}} \frac{1}{2} [(\partial_x a)^2 + 2h\partial_x \alpha \partial_x a + (\partial_x b)^2] - \frac{1}{2\sigma + 2} (a^2 + b^2)^{\sigma+1} \left(1 + (\sigma+1) \frac{2ha\alpha}{a^2 + b^2}\right) + o(h) \\ E(a, b + h\beta) &= \int_{\mathbb{R}} \frac{1}{2} [(\partial_x a)^2 + 2h\partial_x \beta \partial_x b + (\partial_x b)^2] - \frac{1}{2\sigma + 2} (a^2 + b^2)^{\sigma+1} \left(1 + (\sigma+1) \frac{2hb\beta}{a^2 + b^2}\right) + o(h).\end{aligned}$$

Rassemblons les termes de  $E(a, b)$  et les termes dépendants de  $h$  pour faire apparaître  $\frac{\delta \tilde{E}}{\delta a}$  et  $\frac{\delta \tilde{E}}{\delta b}$  :

$$\begin{aligned}E(a + h\alpha, b) &= E(a, b) + h \int_{\mathbb{R}} \left(-\partial_x^2 a - a(a^2 + b^2)^\sigma\right) \alpha + o(h) = E(a, b) + \left\langle \frac{\delta \tilde{E}}{\delta a}, \alpha \right\rangle_{L^2} + o(h) \\ E(a, b + h\beta) &= \tilde{E}(a, b) + h \int_{\mathbb{R}} \left(-\partial_x^2 b - b(a^2 + b^2)^\sigma\right) \beta + o(h) = E(a, b) + \left\langle \frac{\delta \tilde{E}}{\delta b}, \beta \right\rangle_{L^2} + o(h).\end{aligned}$$

On a donc :

$$\begin{aligned}\frac{\delta \tilde{E}}{\delta a} &= -\partial_x^2 a - a(a^2 + b^2)^\sigma \\ \frac{\delta \tilde{E}}{\delta b} &= -\partial_x^2 b - b(a^2 + b^2)^\sigma.\end{aligned}\tag{3.8}$$

On peut donc réécrire notre système (3.7) comme suit

$$\begin{aligned}\partial_t a &= \frac{\delta \tilde{E}}{\delta b} \\ \partial_t b &= -\frac{\delta \tilde{E}}{\delta a}.\end{aligned}\tag{3.9}$$

### 3.3.2.2 Discrétisation de la variation de l'énergie

Maintenant, il faut créer un schéma consistant qui vérifie la conservation de l'énergie. En continu, on a :

$$\frac{d}{dt} \int_{\mathbb{R}} \tilde{E}(a, b) = \int_{\mathbb{R}} \frac{\delta \tilde{E}}{\delta a} \partial_t a + \frac{\delta \tilde{E}}{\delta b} \partial_t b.\tag{3.10}$$

On fixe un entier  $J > 0$  correspondant au nombre de mailles en espace. On définit  $J + 1$  points spatiaux qui seront les noeuds du maillage :  $x_j$  pour  $j \in \llbracket 0, J \rrbracket$ . On suppose pour simplifier qu'ils sont équidistants de même pas d'espace avec  $L$  la longueur du maillage étudié. Nous avons donc :

$$x_j = j\Delta x \quad \forall j \in \llbracket 0, J \rrbracket \quad \text{avec } \Delta x = L/J.$$

On se fixe maintenant un nombre maximal d'itérations temporelles  $N > 0$  ainsi que des temps intermédiaires  $t^n = n\Delta t$  tous séparés du même pas de temps  $\Delta t$  défini par  $\Delta t = \frac{T}{N}$  où  $T$  est la durée d'étude de notre fonction. Nous avons donc :

$$t^n = n\Delta t \quad \forall n \in \llbracket 0, N \rrbracket \quad \text{avec } \Delta t = \frac{T}{N}.$$

On cherche à calculer une valeur approchée des solutions  $u$ ,  $a$  et  $b$  en chaque noeud  $(t^n, x_j)$ . On notera  $v_j^n$ ,  $a_j^n$  et  $b_j^n$  ces solutions approchées de  $u(t^n, x_j)$ ,  $a(t^n, x_j)$  et  $b(t^n, x_j)$ . De plus supposons que notre fonction est périodique et donc que notre approximation assure la condition suivante : Pour tout  $n$ ,  $v_j^n = v_{j\pm J}^n$ ,  $a_j^n = a_{j\pm J}^n$  et  $b_j^n = b_{j\pm J}^n$ . Notons  $V^n = (v_0^n, \dots, v_J^n)$ ,  $a^n = (a_0^n, \dots, a_J^n)$  et  $b^n = (b_0^n, \dots, b_J^n)$ . En discrétisant (3.10), on obtient :

$$\sum_{j=0}^J \left[ \tilde{E}(a_j^{n+1}, b_j^{n+1}) - \tilde{E}(a_j^n, b_j^n) \right] \frac{\Delta x}{\Delta t} = \sum_{j=0}^J \left[ \left( \frac{\delta \tilde{E}}{\delta a} \right)_j^n (a_j^{n+1} - a_j^n) + \left( \frac{\delta \tilde{E}}{\delta b} \right)_j^n (b_j^{n+1} - b_j^n) \right] \frac{\Delta x}{\Delta t}.\tag{3.11}$$

Il ne reste plus qu'à trouver  $\left( \frac{\delta \tilde{E}}{\delta a} \right)_j^n$  et  $\left( \frac{\delta \tilde{E}}{\delta b} \right)_j^n$  vérifiant la conservation de l'énergie. Prenons la discrétisation de  $\tilde{E}$  suivante :

$$\tilde{E}(a_j^n, b_j^n) = \frac{1}{2} \left[ \left( \delta_c^{(1)} a^n \right)_j^2 + \left( \delta_c^{(1)} b^n \right)_j^2 \right] - \frac{1}{2\sigma + 2} \left[ (a_j^n)^2 + (b_j^n)^2 \right]^{\sigma+1}.$$

Tout d'abord nous remarquons que cette expression discrète est bien consistante avec l'expression continue (DENLS) page 41. En remplaçant cette discrétisation dans (3.11), on peut en déduire

l'équation suivante :

$$\sum_{j=0}^J \left[ \tilde{E}(a_j^{n+1}, b_j^{n+1}) - \tilde{E}(a_j^n, b_j^n) \right] \frac{\Delta x}{\Delta t} = \sum_{j=0}^J \frac{1}{2} \left( \left( \delta_c^{(1)} a^{n+1} \right)_j^2 - \left( \delta_c^{(1)} a^n \right)_j^2 + \left( \delta_c^{(1)} b^{n+1} \right)_j^2 - \left( \delta_c^{(1)} b^n \right)_j^2 \right) - \frac{1}{2\sigma + 2} \left( \left( (a_j^{n+1})^2 + (b_j^{n+1})^2 \right)^{\sigma+1} - \left( (a_j^n)^2 + (b_j^n)^2 \right)^{\sigma+1} \right) \frac{\Delta x}{\Delta t}.$$

Les derniers termes peuvent s'écrire sous forme d'une somme en appliquant la fameuse identité géométrique.

$$\begin{aligned} \left( (a_j^{n+1})^2 + (b_j^{n+1})^2 \right)^{\sigma+1} - \left( (a_j^n)^2 + (b_j^n)^2 \right)^{\sigma+1} &= \left( (a_j^{n+1})^2 - (a_j^n)^2 + (b_j^{n+1})^2 - (b_j^n)^2 \right) \\ &\quad \sum_{k=0}^{\sigma} \left( (a_j^{n+1})^2 + (b_j^{n+1})^2 \right)^{\sigma-k} \left( (a_j^n)^2 + (b_j^n)^2 \right)^k \\ &= \left( (a_j^{n+1} - a_j^n) (a_j^{n+1} + a_j^n) + (b_j^{n+1} - b_j^n) (b_j^{n+1} + b_j^n) \right) \\ &\quad \sum_{k=0}^{\sigma} \left( (a_j^{n+1})^2 + (b_j^{n+1})^2 \right)^{\sigma-k} \left( (a_j^n)^2 + (b_j^n)^2 \right)^k. \end{aligned} \quad (3.12)$$

De même nous pouvons réécrire les premiers termes mais cette fois ci en commençant par utiliser la propriété d'antisymétrie (AS) :

$$\begin{aligned} \sum_{j=0}^J \left( \delta_c^{(1)} a^{n+1} \right)_j^2 - \left( \delta_c^{(1)} a^n \right)_j^2 + \left( \delta_c^{(1)} b^{n+1} \right)_j^2 - \left( \delta_c^{(1)} b^n \right)_j^2 &= \sum_{j=0}^J \left[ \left( \delta_c^{(1)} \right)^2 a^n \right]_j a_j^n - \left[ \left( \delta_c^{(1)} \right)^2 a^{n+1} \right]_j a_j^{n+1} \\ &\quad + \left[ \left( \delta_c^{(1)} \right)^2 b^n \right]_j b_j^n - \left[ \left( \delta_c^{(1)} \right)^2 b^{n+1} \right]_j b_j^{n+1}. \end{aligned}$$

Il ne reste plus qu'à utiliser le lemme (L) pour obtenir :

$$\begin{aligned} \sum_{j=0}^J \left( \delta_c^{(1)} a^{n+1} \right)_j^2 - \left( \delta_c^{(1)} a^n \right)_j^2 + \left( \delta_c^{(1)} b^{n+1} \right)_j^2 - \left( \delta_c^{(1)} b^n \right)_j^2 &= \sum_{j=0}^J \left[ \left( \delta_c^{(1)} \right)^2 (a^{n+1} + a^n) \right]_j (a_j^n - a_j^{n+1}) \\ &\quad + \left[ \left( \delta_c^{(1)} \right)^2 (b^{n+1} + b^n) \right]_j (b_j^n - b_j^{n+1}). \end{aligned} \quad (3.13)$$

Finalement, en regroupant les termes en facteur de  $(a_j^{n+1} - a_j^n)$  et ceux en facteur de  $(a_j^{n+1} - a_j^n)$  dans (3.12) et (3.13), on retrouve (3.11). On peut donc déduire les deux variations recherchées :

$$\begin{aligned} \left( \frac{\delta \tilde{E}}{\delta a} \right)_j^n &= -\frac{1}{2} \left( \left( \delta_c^{(1)} \right)^2 (a^{n+1} + a^n) \right)_j - \frac{1}{2\sigma + 2} (a_j^{n+1} + a_j^n) \sum_{k=0}^{\sigma} \left[ (a_j^{n+1})^2 + (b_j^{n+1})^2 \right]^{\sigma-k} \left[ (a_j^n)^2 + (b_j^n)^2 \right]^k \\ \left( \frac{\delta \tilde{E}}{\delta b} \right)_j^n &= -\frac{1}{2} \left( \left( \delta_c^{(1)} \right)^2 (b^{n+1} + b^n) \right)_j - \frac{1}{2\sigma + 2} (b_j^{n+1} + b_j^n) \sum_{k=0}^{\sigma} \left[ (a_j^{n+1})^2 + (b_j^{n+1})^2 \right]^{\sigma-k} \left[ (a_j^n)^2 + (b_j^n)^2 \right]^k. \end{aligned}$$

### 3.3.2.3 Consistance de $\left(\frac{\delta \tilde{E}}{\delta a}\right)_j^n$ et $\left(\frac{\delta \tilde{E}}{\delta b}\right)_j^n$

Vérifions que nos deux variations sont bien consistantes avec les expressions continues (3.8) page 42. En remplaçant  $a_j^n$  par  $a(t^n, x_j)$  (idem pour  $b_j^n$  et  $v_j^n$ ) dans l'expression des variations de la densité calculée juste avant, nous trouvons

$$\begin{aligned} \left(\frac{\delta \tilde{E}}{\delta a}\right)_j^n &= -\frac{1}{2}\partial_x^2(2a) - \frac{1}{2\sigma+2}(2a) \sum_{k=0}^{\sigma} [(a^2 + b^2)]^{\sigma-k} [(a^2 + b^2)]^k + O(\Delta x^6 + \Delta t) \\ &= -\partial_x^2 a - \frac{a}{\sigma+1}(\sigma+1)(a^2 + b^2)^{\sigma} + O(\Delta x^6 + \Delta t) \\ &= -\partial_x^2 a - a(a^2 + b^2)^{\sigma} + O(\Delta x^6 + \Delta t). \end{aligned}$$

De même on obtient :

$$\begin{aligned} \left(\frac{\delta \tilde{E}}{\delta b}\right)_j^n &= -\frac{1}{2}\partial_x^2(2b) - \frac{1}{2\sigma+2}(2b) \sum_{k=0}^{\sigma} [(a^2 + b^2)]^{\sigma-k} [(a^2 + b^2)]^k + O(\Delta x^6 + \Delta t) \\ &= -\partial_x^2 b - \frac{b}{\sigma+1}(\sigma+1)(a^2 + b^2)^{\sigma} + O(\Delta x^6 + \Delta t) \\ &= -\partial_x^2 b - b(a^2 + b^2)^{\sigma} + O(\Delta x^6 + \Delta t). \end{aligned}$$

Ces expressions discrètes sont bien consistantes d'ordre 1 en temps et 6 en espace par rapport aux expressions continues.

### 3.3.2.4 Schéma compact

Afin de simplifier les notations, nous supposons dans ce paragraphe que toutes les opérations entre vecteurs se font terme à terme. Posons  $U^n = \begin{pmatrix} a^n \\ b^n \end{pmatrix}$  et  $U_j^n = \begin{pmatrix} a_j^n \\ b_j^n \end{pmatrix}$ , la discrétisation des équations (3.9) nous donne le schéma suivant

$$\begin{aligned} \frac{U_j^{n+1} - U_j^n}{\Delta t} &= \begin{pmatrix} \left(\frac{\delta \tilde{E}}{\delta b}\right)_j^n \\ -\left(\frac{\delta \tilde{E}}{\delta a}\right)_j^n \end{pmatrix} \\ &= \begin{pmatrix} -\frac{1}{2}((\delta_c^{(1)})^2(b^{n+1} + b^n))_j - \frac{1}{2\sigma+2}(b_j^{n+1} + b_j^n) \sum_{k=0}^{\sigma} [(a_j^{n+1})^2 + (b_j^{n+1})^2]^{\sigma-k} [(a_j^n)^2 + (b_j^n)^2]^k \\ \frac{1}{2}((\delta_c^{(1)})^2(a^{n+1} + a^n))_j + \frac{1}{2\sigma+2}(a_j^{n+1} + a_j^n) \sum_{k=0}^{\sigma} [(a_j^{n+1})^2 + (b_j^{n+1})^2]^{\sigma-k} [(a_j^n)^2 + (b_j^n)^2]^k \end{pmatrix}. \end{aligned}$$

En écrivant notre schéma avec des vecteurs complexes, c'est-à-dire que l'on ne sépare plus les termes réels et les termes imaginaires et que l'on travaille avec des vecteurs de taille  $J+1$  et non plus de taille  $2J+2$ , on peut retrouver le schéma proposé dans l'article [8]. De notre côté, nous garderons la séparation entre les termes réels et les termes imaginaires. Nous pouvons réécrire cette équation sous forme matricielle en fonction de  $U$ . En effet, on a  $\begin{pmatrix} b^n \\ a^n \end{pmatrix} = \begin{pmatrix} 0 & I_J \\ I_J & 0 \end{pmatrix} \begin{pmatrix} a^n \\ b^n \end{pmatrix} = \begin{pmatrix} 0 & I_J \\ I_J & 0 \end{pmatrix} U^n$ . Pour la suite, notons  $S(U^{n+1}) = \sum_{k=0}^{\sigma} [(\begin{smallmatrix} I_J & I_J \end{smallmatrix})(U^{n+1})]^{\sigma-k} [(\begin{smallmatrix} I_J & I_J \end{smallmatrix})(U^n)]^k$ , ce qui

nous permet d'obtenir l'équation matricielle qui suit :

$$\begin{aligned} \frac{U^{n+1} - U^n}{\Delta t} &= \begin{pmatrix} \left(\frac{\delta \tilde{E}}{\delta b}\right)_j^n \\ -\left(\frac{\delta \tilde{E}}{\delta a}\right)_j^n \end{pmatrix} \\ &= \begin{pmatrix} I_J & 0 \\ 0 & -I_J \end{pmatrix} \left[ -\frac{1}{2} \begin{pmatrix} (\delta_c^{(1)})^2 & 0 \\ 0 & (\delta_c^{(1)})^2 \end{pmatrix} \begin{pmatrix} 0 & I_J \\ I_J & 0 \end{pmatrix} (U^{n+1} + U^n) \right] \\ &\quad - \begin{pmatrix} I_J & 0 \\ 0 & -I_J \end{pmatrix} \left[ \frac{1}{2\sigma + 2} \begin{pmatrix} 0 & I_J \\ I_J & 0 \end{pmatrix} (U^{n+1} + U^n) \begin{pmatrix} S(U^{n+1}) \\ S(U^{n+1}) \end{pmatrix} \right] \end{aligned}$$

où la dernière opération est une opération terme à terme. Afin de déterminer le vecteur  $U^{n+1}$  en fonction du vecteur  $U^n$ , il convient d'utiliser, là aussi, la méthode de Newton-Raphson. Pour cela, définissons la fonction  $F_n$  comme suit où  $x$  est un vecteur de taille  $2J + 2$ .

$$F_n(x) = x - U^n + \begin{pmatrix} I_J & 0 \\ 0 & -I_J \end{pmatrix} \left[ \frac{\Delta t}{2} \begin{pmatrix} (\delta_c^{(1)})^2 & 0 \\ 0 & (\delta_c^{(1)})^2 \end{pmatrix} \begin{pmatrix} 0 & I_J \\ I_J & 0 \end{pmatrix} (x + U^n) + \frac{\Delta t}{2\sigma + 2} \begin{pmatrix} 0 & I_J \\ I_J & 0 \end{pmatrix} (x + U^n) \begin{pmatrix} S(x) \\ S(x) \end{pmatrix} \right]$$

Pour appliquer la méthode de Newton-Raphson, il ne reste plus qu'à déterminer la jacobienne de  $F_n$ . La jacobienne n'étant pas facile à déterminer, notons :

$$\begin{aligned} P(x_a, x_b) &= \sum_{k=0}^{\sigma} \left[ (x_a)^2 + (x_b)^2 \right]^{\sigma-k} \left[ (a_j^n)^2 + (b_j^n)^2 \right]^k, \\ \partial_{x_a} P(x_a, x_b) &= \sum_{k=0}^{\sigma-1} (\sigma - k) \left[ (x_a)^2 + (x_b)^2 \right]^{\sigma-k-1} 2x_a \left[ (a_j^n)^2 + (b_j^n)^2 \right]^k, \\ \partial_{x_b} P(x_a, x_b) &= \sum_{k=0}^{\sigma-1} (\sigma - k) \left[ (x_a)^2 + (x_b)^2 \right]^{\sigma-k-1} 2x_b \left[ (a_j^n)^2 + (b_j^n)^2 \right]^k. \end{aligned}$$

Ce qui nous permet de réécrire  $F_n$  tel quel

$$F_n(x) = \begin{pmatrix} F_0^1(x_a, x_b) \\ \vdots \\ F_J^1(x_a, x_b) \\ F_0^2(x_a, x_b) \\ \vdots \\ F_J^2(x_a, x_b) \end{pmatrix}$$

où  $x = \begin{pmatrix} x_a \\ x_b \end{pmatrix}$  avec  $x_a$  et  $x_b$  de taille  $J + 1$ . Posons une série de notations nous permettant de simplifier l'écriture de la jacobienne :

$$\begin{aligned} F_j^1(x_a, x_b) &= x_a - a_j^n + \frac{\Delta t}{2} \left( \left( \delta_c^{(1)} \right)^2 (x_b + b_j^n) \right)_j + \frac{\Delta t}{2\sigma + 2} (x_{b_j} + b_j^n) P(x_a, x_b), \\ F_j^2(x_a, x_b) &= x_b - b_j^n - \frac{\Delta t}{2} \left( \left( \delta_c^{(1)} \right)^2 (x_a + a_j^n) \right)_j - \frac{\Delta t}{2\sigma + 2} (x_{a_j} + a_j^n) P(x_a, x_b), \end{aligned}$$

$$\begin{aligned}
\frac{\partial F_k^1}{\partial x_{a_l}}(x_a, x_b) &= \begin{cases} 1 + \frac{\Delta t}{2\sigma+2} [(x_b + b^n) \partial_{x_a} P(x_a, x_b)]_k & \text{si } k = l, \\ 0 & \text{sinon.} \end{cases} \\
\frac{\partial F_k^1}{\partial x_{b_l}}(x_a, x_b) &= \begin{cases} \frac{\Delta t}{2} \left( \delta_c^{(1)} \right)_{k,k}^2 + \frac{\Delta t}{2\sigma+2} [P(x_a, x_b) + (x_b + b^n) \partial_{x_b} P(x_a, x_b)]_k & \text{si } k = l, \\ \frac{\Delta t}{2} \left( \delta_c^{(1)} \right)_{k,l}^2 & \text{sinon.} \end{cases} \\
\frac{\partial F_k^2}{\partial x_{a_l}}(x_a, x_b) &= \begin{cases} -\frac{\Delta t}{2} \left( \delta_c^{(1)} \right)_{k,k}^2 - \frac{\Delta t}{2\sigma+2} [P(x_a, x_b) + (x_a + a^n) \partial_{x_b} P(x_a, x_b)]_k & \text{si } k = l, \\ -\frac{\Delta t}{2} \left( \delta_c^{(1)} \right)_{k,l}^2 & \text{sinon.} \end{cases} \\
\frac{\partial F_k^2}{\partial x_{b_l}}(x_a, x_b) &= \begin{cases} 1 - \frac{\Delta t}{2\sigma+2} [(x_a + a^n) \partial_{x_a} P(x_a, x_b)]_k & \text{si } k = l, \\ 0 & \text{sinon.} \end{cases}
\end{aligned}$$

Posons aussi les quatre matrices carrées, de taille  $J + 1$ , suivantes :

$$\begin{aligned}
M &= \begin{pmatrix} (b^n + x_b)_0 \partial_{x_a} P(x_a, x_b)_0 & & (0) \\ & \ddots & \\ (0) & & (b^n + x_b)_J \partial_{x_a} P(x_a, x_b)_J \end{pmatrix} \\
G &= \begin{pmatrix} (a^n + x_a)_0 \partial_{x_b} P(x_a, x_b)_0 & & (0) \\ & \ddots & \\ (0) & & (a^n + x_a)_J \partial_{x_b} P(x_a, x_b)_J \end{pmatrix} \\
N &= \begin{pmatrix} P(x_a, x_b)_0 + (b^n + x_b)_0 \partial_{x_b} P(x_a, x_b)_0 & & (0) \\ & \ddots & \\ (0) & & P(x_a, x_b)_J + (b^n + x_b)_J \partial_{x_b} P(x_a, x_b)_J \end{pmatrix} \\
E &= \begin{pmatrix} P(x_a, x_b)_0 + (a^n + x_a)_0 \partial_{x_a} P(x_a, x_b)_0 & & (0) \\ & \ddots & \\ (0) & & P(x_a, x_b)_J + (a^n + x_a)_J \partial_{x_a} P(x_a, x_b)_J \end{pmatrix}
\end{aligned}$$

Ce qui nous permet d'écrire la jacobienne de  $F_n$  beaucoup plus facilement.

$$J_n(x) = \left( \begin{array}{c|c} I_J + \frac{\Delta t}{2\sigma+2} M & \frac{\Delta t}{2} \left( \delta_c^{(1)} \right)^2 + \frac{\Delta t}{2\sigma+2} N \\ \hline -\frac{\Delta t}{2} \left( \delta_c^{(1)} \right)^2 - \frac{\Delta t}{2\sigma+2} E & I_J - \frac{\Delta t}{2\sigma+2} G \end{array} \right)$$

Pour simplifier les notations, nous n'avons pas tenus compte ici des conditions aux bords. Dans le code, les conditions aux bords sont périodiques. Les matrices implémentées sont donc de taille  $J$  avec des coefficients non nuls en haut à droite et en bas à gauche.

Ici, on se place dans les conditions (C1). Comme nous cherchons la solution dynamique de (NLS), nous étudierons la solution (SNLS). Finalement, à l'aide du code mis en annexe, nous pouvons tracer le soliton avec les variables :  $L = 30$ ,  $J = 200$ ,  $T = 40$  et  $N = 2000$ .

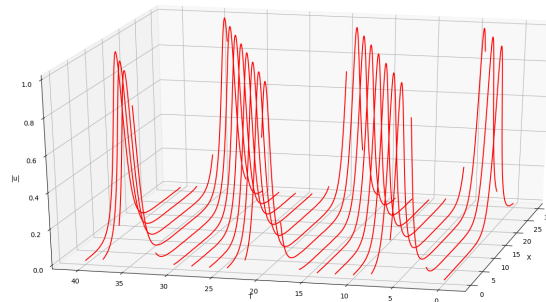


FIGURE 25 – Evolution du soliton avec  $c = 2$ ,  $v = 0.5$ ,  $L = 30$ ,  $J = 200$ ,  $T = 40$  et  $N = 2000$

Ainsi, le schéma compact nous a permis de retrouver le soliton de (NLS) dynamique contrairement à la méthode de tir qui permet de modéliser seulement l'enveloppe.

On peut essayer de vérifier, de plus, l'ordre de convergence du schéma compact. Théoriquement, le schéma est d'ordre 1 en temps et 6 en espace. Ainsi, notre erreur est un  $O(\Delta t + \Delta x^6)$ . Si l'on prend  $\Delta t = O(\Delta x)$ , l'erreur devient  $O(\Delta t + \Delta x^6) = O(\Delta x)$ . Il faut donc prendre  $\Delta t = O(\Delta x^6)$  pour retrouver l'ordre 6 en espace indiqué dans [11]. Cependant, cela est totalement infaisable sur un ordinateur personnel. Les deux figures suivantes obtenues après de longues minutes d'attente nous montre le comportement décrit ci-dessous. Pour cela, on pose  $T = 2$  et  $J = 20^m$  pour  $m$  allant de 1 à 5.

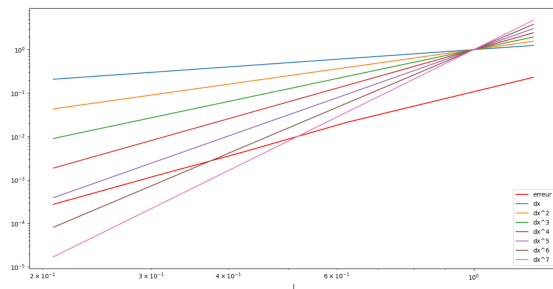


FIGURE 26 – Erreur en fonction de  $J$  lorsque  $\Delta x = \Delta t$

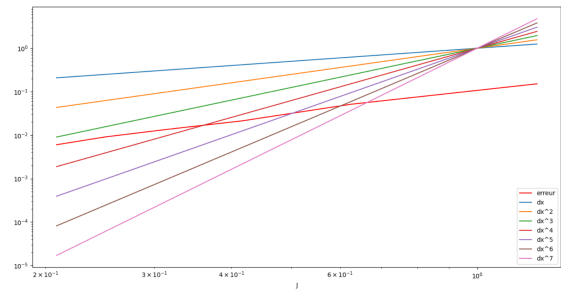


FIGURE 27 – Erreur en fonction de  $J$  lorsque  $\Delta x = \Delta t^2$



### 3.3.2.5 Conservation de l'énergie discrète

Désormais, vérifions que l'énergie discrète de (NLS) est bien conservée avec les variations discrètes trouvées. Pour cela nous utiliseront la discrétisation du schéma (3.9) :

$$\begin{aligned}
\sum_{j=0}^J \left[ \tilde{E}(a_j^{n+1}, b_j^{n+1}) - \tilde{E}(a_j^n, b_j^n) \right] \frac{\Delta x}{\Delta t} &= \sum_{j=0}^J \left[ \left( \frac{\delta \tilde{E}}{\delta a} \right)_j^n (a_j^{n+1} - a_j^n) + \left( \frac{\delta \tilde{E}}{\delta b} \right)_j^n (b_j^{n+1} - b_j^n) \right] \frac{\Delta x}{\Delta t} \\
&= \sum_{j=0}^J \left[ \left( \frac{\delta \tilde{E}}{\delta a} \right)_j^n \left( \frac{a_j^{n+1} - a_j^n}{\Delta t} \right) + \left( \frac{\delta \tilde{E}}{\delta b} \right)_j^n \left( \frac{b_j^{n+1} - b_j^n}{\Delta t} \right) \right] \Delta x \\
&= \sum_{j=0}^J \left[ \left( \frac{\delta \tilde{E}}{\delta a} \right)_j^n \left( \frac{\delta \tilde{E}}{\delta b} \right)_j^n - \left( \frac{\delta \tilde{E}}{\delta b} \right)_j^n \left( \frac{\delta \tilde{E}}{\delta a} \right)_j^n \right] \Delta x \\
&= 0.
\end{aligned}$$

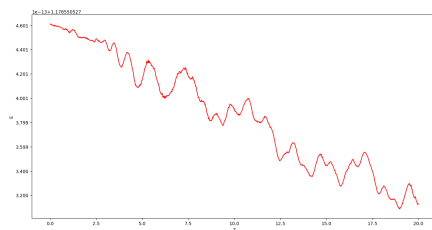


FIGURE 28 – Energie de l'équation de NLS

Nous pouvons voir que l'énergie discrète de (NLS) est quasiment constante avec une erreur d'ordre  $10^{-13}$ . Il faut faire attention à l'échelle pour se rendre compte de la conservation. L'erreur étant négligeable, on peut donc conclure que le schéma conserve l'énergie discrète.

### 3.3.2.6 Conservation de la norme $L^2$

En prenant la notation  $u = a + ib$ , nous pouvons réécrire la norme  $L^2$  comme suit :

$$\frac{d}{dt} L(a, b) = \frac{d}{dt} \int_R a^2 + b^2.$$

La discrétisation de la dérivée la norme  $L^2$  ci-dessous nous amène facilement à sa conservation :

$$\begin{aligned}
\sum_{j=0}^J \left[ (a_j^{n+1})^2 + (b_j^{n+1})^2 - (a_j^n)^2 - (b_j^n)^2 \right] \frac{\Delta x}{\Delta t} &= \sum_{j=0}^J \left[ \left( \frac{a_j^{n+1} - a_j^n}{\Delta t} \right) (a_j^{n+1} + a_j^n) + \left( \frac{b_j^{n+1} - b_j^n}{\Delta t} \right) (b_j^{n+1} + b_j^n) \right] \Delta x \\
&= \sum_{j=0}^J \left[ \left( \frac{\delta \tilde{E}}{\delta b} \right)_j^n (a_j^{n+1} + a_j^n) - \left( \frac{\delta \tilde{E}}{\delta a} \right)_j^n (b_j^{n+1} + b_j^n) \right] \Delta x \\
&= 0.
\end{aligned}$$

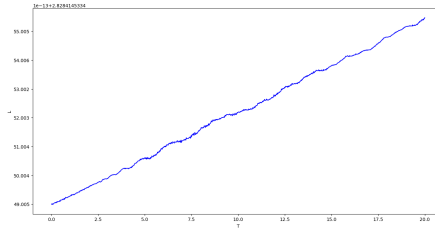


FIGURE 29 – Norme  $L^2$  de l'équation de NLS

Nous pouvons voir que la norme  $L^2$  de (NLS) est constante avec une erreur d'ordre  $10^{-13}$ . L'erreur étant négligeable, on peut donc conclure que le schéma conserve la norme  $L^2$ .

### 3.3.2.7 Conservation du moment

La preuve analytique de la conservation du moment discret est beaucoup plus compliquée que celle de la norme  $L^2$  et celle de l'énergie. On peut supposer la conservation du moment vérifiée par la méthode numérique.

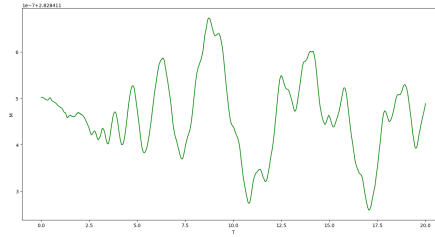


FIGURE 30 – Moment de l'équation de NLS

Nous pouvons voir que le moment discret de (NLS) est constant avec une erreur d'ordre  $10^{-7}$ . L'erreur étant négligeable, on peut donc conclure que le schéma conserve le moment.

## 4 Conclusion

Nous avons pu voir au travers de ce mémoire que le choix du schéma est important pour modéliser ce que l'on veut. En effet, si l'on veut modéliser un soliton, il faut choisir un certain schéma. On peut revenir à l'équation de KdV où la méthode de Heun ne convient absolument pas, ni pour capter le soliton ni pour conserver l'hamiltonien. Le schéma compact sera à privilégier dans ce cas. De même, on préférera utiliser la méthode de tir pour modéliser uniquement l'enveloppe du soliton, nous donnant l'avantage de ne pas avoir à connaître une des conditions initiales. Cependant, comme nous l'avons vu la méthode de tir ne reste pas optimale à partir d'un certain  $L$  pour (KdV). Quant à l'adaptation du schéma compact à (NLS), tout convient parfaitement et il est même symplectique.



## A Résolution de l'équation de NLS en posant $u(t, x) = \phi(x)e^{i\omega t}$

Cette annexe propose une alternative au calcul du soliton pour l'équation de NLS. Cette alternative est plus simple mais moins générale que la méthode que nous avons choisie de suivre dans le mémoire.

Comme nous cherchons un soliton et que l'équation de NLS est complexe, on peut chercher des solutions de la forme  $u(t, x) = \phi(x)e^{i\omega t}$  avec  $\phi : \mathbb{R} \rightarrow \mathbb{R}$  [12] où la fonction  $\phi$  est localisée et strictement positive. En injectant  $u$  dans l'équation de NLS, on obtient :

$$i\phi(x)e^{i\omega t}i\omega + \phi''(x)e^{i\omega t} + |\phi(x)|^{2\sigma}\phi(x)e^{i\omega t} = 0.$$

Or  $\phi$  est à valeurs réelles donc  $|\phi(x)|^{2\sigma} = (\phi(x))^{2\sigma}$  et l'équation devient donc

$$\phi''(x) + \phi(x)^{2\sigma+1} - \phi(x)\omega = 0.$$

En multipliant par  $\phi'$  et intégrant la dernière équation obtenue, on a :

$$\frac{1}{2}(\phi'(x))^2 + \frac{1}{2\sigma+2}\phi^{2\sigma+2}(x) - \frac{1}{2}\phi^2(x)\omega = A$$

où  $A$  est une constante réelle. Comme  $\phi$  est localisée, en faisant tendre  $x$  vers l'infini, on obtient :

$$A = 0.$$

Donc on aboutit à

$$\frac{1}{2}(\phi'(x))^2 + \frac{1}{2\sigma+2}\phi^{2\sigma+2}(x) - \frac{1}{2}\phi^2(x)\omega = 0$$

que l'on peut réécrire comme

$$(\phi'(x))^2 = \phi^2(x)\omega - \frac{1}{\sigma+1}\phi^{2\sigma+2}(x).$$

$(\phi')^2$  est positive donc  $\phi^2\omega - \frac{1}{\sigma+1}\phi^{2\sigma+2}$  aussi. On en déduit donc :

$$\phi' = \pm \sqrt{\phi^2\omega - \frac{1}{\sigma+1}\phi^{2\sigma+2}} = \pm |\phi| \sqrt{\omega - \frac{1}{\sigma+1}\phi^{2\sigma}}.$$

Toujours d'après nos hypothèses sur le soliton, notre solution est strictement positive donc nous pouvons appliquer la séparation des variables :

$$\frac{d\phi}{\phi \sqrt{\omega - \frac{1}{\sigma+1}\phi^{2\sigma}}} = \pm dx \tag{A.1}$$

Posons  $\psi = \frac{1}{\sqrt{\phi}}$  alors  $d\psi = -\frac{d\phi}{2\phi\sqrt{\phi}}$  Donc (A.1) devient :

$$-\frac{2d\psi\sqrt{\phi}}{\phi \sqrt{\omega - \frac{1}{\sigma+1}\phi^{2\sigma}}} = \pm dx,$$

$$\frac{2d\psi\sqrt{\phi}}{\frac{\phi^\sigma}{\sqrt{\sigma+1}} \sqrt{\frac{\omega(\sigma+1)}{\phi^{2\sigma}} - 1}} = \mp dx,$$

or  $\psi^\sigma = \frac{1}{\phi^{\frac{\sigma}{2}}}$ , on obtient donc

$$\frac{d\psi\psi^{2\sigma-1}}{\sqrt{\omega(\sigma+1)\psi^{4\sigma}-1}}2\sqrt{\sigma+1} = \mp dx. \quad (\text{A.2})$$

Comme on l'a déjà vu avec KdV,  $\text{argcosh}'(u) = \frac{u'}{\sqrt{u^2-1}}$ . Donc  $\text{argcosh}'(u^{2\sigma}) = \frac{2\sigma u^{2\sigma-1}u'}{\sqrt{u^{4\sigma}-1}}$ , ainsi on peut réécrire (A.2) :

$$\frac{2\sqrt{\sigma+1}d\text{argcosh}(\sqrt{\omega(\sigma+1)}\psi^{2\sigma})}{2\sigma\sqrt{\omega(\sigma+1)}} = \mp dx.$$

En primitivant la dernière équation, on obtient :

$$\frac{1}{\sigma\sqrt{\omega}} \text{argcosh}(\sqrt{\omega(\sigma+1)}\psi^{2\sigma}) = \mp x + B$$

où  $B$  est une constante réelle. Ainsi on a :

$$\psi(x) = \left( \frac{\cosh(\sigma\sqrt{\omega}(\mp x + B))}{\sqrt{\omega(\sigma+1)}} \right)^{\frac{1}{2\sigma}}. \quad (\text{A.3})$$

Or  $\psi = \frac{1}{\sqrt{\phi}}$ , donc

$$\phi(x) = \left( \frac{\omega(\sigma+1)}{\cosh^2(\sigma\sqrt{\omega}(\mp x + B))} \right)^{\frac{1}{2\sigma}}. \quad (\text{A.4})$$

Afin de simplifier la suite, on se place de façon à ce que  $\phi$  possède un maximum en  $x = 0$  et ainsi qu'elle soit symétrique par rapport à l'axe des ordonnées. Pour se faire, il faut que  $B = 0$ . Comme, de plus, le cosinus hyperbolique est une fonction paire, on obtient [3] :

$$u(t, x) = \left( \frac{\omega(\sigma+1)}{\cosh^2(\sigma\sqrt{\omega}x)} \right)^{\frac{1}{2\sigma}} e^{i\omega t}. \quad (\text{A.5})$$

En posant  $\tilde{\omega} = (c^2 - 2cv)/4$  dans l'expression (SNLS), nous obtenons à la place de (SNLS) :

$$u(t, x) = \left( \frac{\tilde{\omega}(\sigma+1)}{\cosh^2(\sigma\sqrt{\tilde{\omega}}(x - ct))} \right)^{\frac{1}{2\sigma}} e^{i\left(\frac{cx}{2} + \tilde{\omega}t - c^2\frac{t}{4}\right)}.$$

En faisant tendre la vitesse  $c$  vers 0, nous retrouvons bien l'expression (A.5) ce qui signifie que l'expression (A.5) correspond à l'enveloppe du soliton sans la dynamique ( $c = 0$ ).

## B Codes

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3 from matplotlib import cm
4 from matplotlib.ticker import LinearLocator, FormatStrFormatter
5 from mpl_toolkits.mplot3d import Axes3D
6 import matplotlib.animation as anim
7 from scipy.optimize import fsolve
```

```

8 from scipy.integrate import odeint
9
10 i_comp=np.complex(0,1)

```

Listing 1 – Entête

## B.1 Méthode de Newton

```

1 def Newton(xn,F,Jac,eps):
2     x=xn
3     while np.linalg.norm(F(x,xn))>eps:
4         x=x-np.dot(np.linalg.inv(Jac(x,xn)),F(x,xn)) #x_(n+1)=x_n-inv(J(xn))*F(xn)
5     return x

```

Listing 2 – Méthode de Newton-Raphson

## B.2 Equation de KdV

### B.2.1 Portrait de phase

```

1 """Definition des variables"""
2 x = np.linspace(-1,3,20)
3 y = np.linspace(-2,2,20)
4 X,Y = np.meshgrid(x,y)
5
6 vitesse=1
7
8
9 """Fonction vectorielle definissant l'EDO"""
10 def G(X, x=0) : #G l'application definissant l'EDO suivante : [u,u']' = [u',c*u-u
    **2/2] = G([u,u'],x)
11     return np.array([X[1],vitesse*X[0]-X[0]**2/2.])
12
13
14 """Traces"""
15 fig,ax=plt.subplots(figsize=(7,7))
16 u,v=G([X,Y])
17 ax.quiver(X,Y,u,v)
18 plt.title('Portrait de phase KdV avec une vitesse egale a 1')
19 plt.grid()
20 plt.show()

```

Listing 3 – Portrait de phase de KdV

### B.2.2 Schéma compact

```

1 """Discretisation spatiale"""
2 L = 50
3 J = 100
4 dx = L/J
5 X = np.linspace(0,L,J+1)
6
7
8 """Discretisation temporelle"""
9 N = 1000
10 dt = 1/50

```

```

11 T = int(N*dt)
12 temps = np.linspace(0,T,N+1)
13
14
15 """Differentes jeux de variables"""
16 alpha = 0                                #(C1)
17 beta = 0
18 a = 3./2.
19 b = -3./5.
20 c = 1./10.
21
22 alpha = 0.5381301488732363                #(C2)
23 beta = 0.066633190123881123
24 a = 1.367577724399269
25 b = 0.8234281701082790
26 c = 0.018520783486686603
27
28
29 """Autres variables"""
30 vitesse = 1.
31
32 u = lambda x,t: 3*vitesse*(1/np.cosh((np.sqrt(vitesse)/2)*((x-vitesse*t)%L-L/2)))
33     **2
34
35 A = np.diag(np.ones(J))+np.diag(np.ones(J-1)*alpha,-1)+np.diag(np.ones(J-1)*alpha
36     ,1)+np.diag(np.ones(J-2)*beta,-2)+np.diag(np.ones(J-2)*beta,2)
37 A[0][J-1] = alpha
38 A[0][J-2] = beta
39 A[1][J-1] = beta
40 A[J-1][0] = alpha
41 A[J-1][1] = beta
42 A[J-2][0] = beta
43
44 B = np.diag(np.ones(J-1)*(-a/(2*dx)), -1)+np.diag(np.ones(J-1)*(a/(2*dx)), 1)+np.
45     diag(np.ones(J-2)*(-b/(4*dx)), -2)+np.diag(np.ones(J-2)*(b/(4*dx)), 2)+np.diag(
46     np.ones(J-3)*(-c/(6*dx)), -3)+np.diag(np.ones(J-3)*(c/(6*dx)), 3)
47 B[0][J-1] = -a/(2*dx)
48 B[0][J-2] = -b/(4*dx)
49 B[0][J-3] = -c/(6*dx)
50 B[1][J-1] = -b/(4*dx)
51 B[1][J-2] = -c/(6*dx)
52 B[2][J-1] = -c/(6*dx)
53 B[J-1][0] = a/(2*dx)
54 B[J-1][1] = b/(4*dx)
55 B[J-1][2] = c/(6*dx)
56 B[J-2][0] = b/(4*dx)
57 B[J-2][1] = c/(6*dx)
58 B[J-3][0] = c/(6*dx)
59
60 A_inv_B = np.matmul(np.linalg.inv(A),B)                                #A^{-1}B
61 A_inv_B3 = np.matmul(np.matmul(A_inv_B,A_inv_B),A_inv_B)              #(A^{-1}B)^{3}
62
63 def F(x,xn):    #F_{n}
64     return x+(dt/6)*np.dot(A_inv_B,(x*x+x*xn+xn*xn))+(dt/2)*np.dot(A_inv_B3,(x+xn)
65     )-xn
66
67 def Jac(x,xn):  #J_{n}
68     return np.diag(np.ones(J))+dt/2*A_inv_B3+dt/6*np.matmul(A_inv_B,np.diag(2*x+xn
69     ))
70
71 """Hamiltonien"""

```



```

67 def H(v):
68     s=0
69     for i in range(np.size(v)):
70         s+=-1/6*(v[i])**3+1/2*(np.dot(A_inv_B,v)[i])**2
71     return s*dx
72
73 Hamil=[]
74
75 """Schema"""
76 eps = 10**(-12) #Precision de Newton-Raphson
77 v0 = u(X[:J],0)
78 v = v0
79 v0_graphe = np.zeros(J+1)
80 v0_graphe[:J] = v
81 v0_graphe[J] = v[0] #Condition periodique
82 V = [v0_graphe]
83 Hamil.append(H(v0))
84
85 for n in range(np.size(temps)-1): #Il reste N termes a calculer car le premier
    terme est connu
86     v =Newton(v,F,Jac,eps)
87     v_graphe = np.zeros(J+1)
88     v_graphe[:J] = v
89     v_graphe[J] = v[0] #Condition periodique
90     V.append(v_graphe)
91     Hamil.append(H(v))
92 V_trace = np.array(V)
93
94
95 """Animation"""
96 fig=plt.figure(2)
97 plt.clf()
98 ax=fig.gca()
99 line1, =ax.plot(X, u(X,0), 'r')
100 line2, =ax.plot(X, V[0], 'b--')
101 ax.set_title('Temps initial')
102 plt.legend(['sol exacte','sol approchee'], loc='best')
103
104
105 """Mise a jour de chaque plot"""
106 def runanimate1(n):
107     line1.set_data(X, u(X,dt*n))
108     ax.set_title('Temps t=' + str('{:.3f}'.format(n*dt)))
109
110 def runanimate2(n):
111     line2.set_data(X, V[n])
112     ax.set_title('Temps t=' + str('{:.3f}'.format(n*dt)))
113
114
115 ani1=anim.FuncAnimation(fig,runanimate1,frames=np.arange(N+1),interval=1,repeat=
    False)
116 ani2=anim.FuncAnimation(fig,runanimate2,frames=np.arange(N+1),interval=1,repeat=
    False)
117 plt.show()
118
119
120 """Afficher espace temps"""
121 fig=plt.figure()
122 X2,Y2=np.meshgrid(X,temps)
123 ax=fig.gca(projection='3d')
124 ax.plot_wireframe(X2,Y2,V_trace,color='r',rstride=100, cstride=0)
125 plt.title("Trace filaire KdV")

```

```

126 ax.set_xlabel('X')
127 ax.set_ylabel('T')
128 ax.set_zlabel('u')
129 plt.tight_layout()
130 plt.show()

```

Listing 4 – Schéma compact pour KdV

### B.2.3 Méthode de Heun pour le schéma compact

```

1  """Discretisation spatiale"""
2  L = 50
3  J = 100
4  dx = L/J
5  X = np.linspace(0,L,J+1)
6
7
8  """Discretisation temporelle"""
9  N = 500
10 dt = 1/50
11 T = int(N*dt)
12 temps = np.linspace(0,T,N+1)
13
14
15 """Fonction caracterisant l'EDO"""
16 def G(x,X): #G(t,V^{n})=-V^{n}(A^{-1}BV^{n})-(A^{-1}B)^{3}V^{n}
17     return -X*np.dot(A_inv_B,X)-np.dot(A_inv_B3,X)
18
19
20 """Hamiltonien"""
21 plt.figure(1)
22 plt.clf()
23 plt.plot(temps, np.array(Hamil))
24
25
26 """Schema"""
27 t = 0
28 v = u(X[:J],0)
29 V = [v]
30 Hamil.append(H(w))
31 for n in range(np.size(temps)-1):
32     k1 = G(t+dt,v)
33     k2 = G(t+dt,v+dt*k1)
34     v = v+dt/2*(k1+k2)
35     V.append(v)
36     Hamil.append(H(v))
37
38
39 """Animation"""
40 fig=plt.figure(2)
41 plt.clf()
42 ax=fig.gca()
43 line1, =ax.plot(X[:J], u(X[:J],0), 'r')
44 line2, =ax.plot(X[:J], V[0], 'b--')
45 ax.set_title('Temps initial')
46 plt.legend(['sol exacte','sol approchee'], loc='best')
47
48
49 """Mise a jour de chaque plot"""
50 def runanimate1(n):

```

```

51     line1.set_data(X[:,J], u(X[:,J],dt*n))
52     ax.set_title('Temps t=' + str('{:.3f}'.format(n*dt)))
53
54 def runanimate2(n):
55     line2.set_data(X[:,J], V[n])
56     ax.set_title('Temps t=' + str('{:.3f}'.format(n*dt)))
57
58 ani1=anim.FuncAnimation(fig,runanimate1,frames=np.arange(N+1),interval=1,repeat=
    False)
59 ani2=anim.FuncAnimation(fig,runanimate2,frames=np.arange(N+1),interval=1,repeat=
    False)
60
61
62 """Afficher espace temps"""
63 fig=plt.figure()
64 X2,Y2=np.meshgrid(X[:,J],temps)
65 ax=fig.gca(projection='3d')
66 ax.plot_wireframe(X2,Y2,np.array(V),color='r',rstride=100, cstride=0)
67 ax.set_xlabel('X')
68 ax.set_ylabel('T')
69 ax.set_zlabel('u')
70 plt.tight_layout()
71 plt.show()

```

Listing 5 – Schéma compact (Heun) KdV

## B.2.4 Méthode de tir pour KdV

```

1  """Discretisation spatiale"""
2  L=50
3  J=40000
4  dx=L/J
5
6
7  """Definitions des variables"""
8  eps=10**(-8)
9
10 a=2
11 b=5
12
13 vitesse=1.
14 u=lambda y: 3*vitesse*(1/np.cosh((np.sqrt(vitesse)/2)*((y))))**2
15
16 def F(x,X):
17     Z=np.zeros(2)
18     Z[0]=X[1]
19     Z[1]=vitesse*X[0]-X[0]**2/2. #- vitesse*alpha +alpha**2/2. +3/2.
20     return Z
21
22
23 """Schema"""
24 n=1
25 while n>0:
26     alpha=(a+b)/2.
27     Y=np.zeros(2)
28     Y[0]=alpha
29     y=0
30     Y_liste=[Y[0]] #u_tilde
31     for i in range (J):
32         k1=K(y,Y)

```

```

33     k2=K(y+dx/2.,Y+dx/2.*k1)
34     k3=K(y+dx/2.,Y+dx/2.*k2)
35     k4=K(y+dx,Y+dx*k3)
36     Y=Y+dx/6.*(k1+2.*k2+2.*k3+k4)
37     Y_liste.append(Y[0])
38     y=y+dx
39     c=0
40     for i in range (len(Y_liste)):
41         # print("Y",Y_liste[i])
42         if Y_liste[i]<0 :    #compteur c
43             c+=1
44             b=alpha
45     if c==0:
46         for j in range(int(100.*len(Y_liste)/101.),len(Y_liste)):
47             if Y_liste[j]>eps :
48                 a=alpha
49                 n=1
50             else :
51                 n=0
52
53
54     """Traces"""
55     plt.figure(1)
56     plt.clf()
57     plt.plot(X,Y_tableau,'b--')
58     plt.plot(X,u(X2),'r')
59     plt.show()

```

Listing 6 – Méthode de tir pour KdV

## B.3 L'équation de NLS

### B.3.1 Portrait de phase

```

1  """Definition des variables"""
2  x = np.linspace(0,2,20)
3  y = np.linspace(-0.5,0.5,20)
4  X,Y = np.meshgrid(x,y)
5
6  sigma = 3
7  c = 2
8  v = 0.5
9
10
11 """Fonction vectorielle definissant l'EDO"""
12 def F(X, x=0) :
13     return np.array([X[1],(c**2-2*c*v)/4*X[0]-X[0]**(2*sigma+1)])
14
15
16 """Traces"""
17 u,v=F([X,Y])
18 ax.quiver(X,Y,u,v)
19 plt.title('Portrait de phase NLS avec une vitesse egale a 1')
20 plt.grid()
21 plt.show()

```

Listing 7 – Portrait de phase de NLS

### B.3.2 Méthode de tir

```
1
2 """Discretisation spatiale"""
3 L = 20
4 J = 2000
5 dx = L/J
6 X = np.linspace(0,L,J+1)
7
8
9 """Definition des variables"""
10 eps=10**(-8)
11 sigma=1
12
13 a=0.9
14 b=2
15
16 u=lambda x,t: ((sigma+1.)*(1./np.cosh(sigma*x))**2)**(1./(2.*sigma))*np.exp(i_comp
    *t)
17
18
19
20 """Hamiltonien"""
21 def Hamil(Y,Ydx):
22     s=0
23     for i in range(len(Y)):
24         s+=(1./2.)*abs(Ydx[i])**2-(1./2.)*(1./(sigma+1.))*abs(Y[i])**2*(sigma+2)
25     return s*dx
26
27 H=[]
28
29
30 """Fonction F"""
31 def F(x,X):
32     Z=np.zeros(2)
33     Z[0]=X[1]
34     Z[1]=(1-(abs(X[0]))**(2*sigma))*X[0]
35     return Z
36
37
38 """Schema"""
39 n=1
40 while n>0:
41     alpha=(a+b)/2.
42     Y=np.zeros(2)
43     Y[0]=alpha #Conditions initiales
44     y=0
45     Y_liste=[Y[0]] #phi_tilde
46     Ydx_liste=[Y[1]] #derivee de phi_tilde
47     for i in range (J):
48         k1=F(y,Y)
49         k2=F(y+dx/2.,Y+dx/2.*k1)
50         k3=F(y+dx/2.,Y+dx/2.*k2)
51         k4=F(y+dx,Y+dx*k3)
52         Y=Y+dx/6.*(k1+2.*k2+2.*k3+k4)
53         Y_liste.append(Y[0])
54         Ydx_liste.append(Y[1])
55         y=y+dx
56     c=0 #compteur c pour savoir si il existe une valeur <0
57     H.append(Hamil(Y_liste,Ydx_liste))
58     for i in range (len(Y_liste)):
```

```

59         if Y_liste[i]<0 :
60             c+=1
61             b=alpha
62
63     if c==0:
64         for j in range(int(100.*len(Y_liste)/101.),len(Y_liste)): #dernieres
valeurs
65             if Y_liste[j]>eps :
66                 a=alpha
67                 n=1
68             else :
69                 n=0          #arret car les deux criteres ont ete satisfaits
70
71
72     """Traces"""
73     plt.figure(1)
74     plt.clf()
75     plt.plot(X,Y_liste,'y.',label='solution approchee')
76     plt.plot(X,u(X,0).real,'r-',label='solution exacte')
77     plt.xlabel('x')
78     plt.ylabel('phi_tilde')
79     plt.legend()
80
81     plt.figure(2)
82     Xhamil=np.linspace(0,len(H),len(H))
83     plt.plot(Xhamil,np.array(H))
84     plt.xlabel('Etape de la dichotomie')
85     plt.ylabel('H')
86     plt.legend()
87
88     plt.show()

```

Listing 8 – Méthode de tir pour l'équation de NLS

### B.3.3 Schéma compact pour l'équation de NLS

```

1  """Discretisation spatiale"""
2  L=30
3  J=200
4  dx=L/J
5  X=np.linspace(0,L,J+1)
6
7
8  """Discretisation temporelle"""
9  N=2000
10 dt=1/50
11 T=int(N*dt)
12 temps=np.linspace(0,T,N+1)
13
14
15 """Hamiltonien"""
16 def H(v):
17     s=0
18     dxv=np.dot(mat3,v)
19     for i in range(J):
20         s+=1./2.*(dxv[i]**2+dxv[i+J]**2)-1./(2.*sigma+2.)*(v[i]**2+v[i+J]**2)**(
sigma+1)
21     return s*dx
22
23 Hamil=[]

```

```

24
25
26 """Norme L_2"""
27 def Norme_L(v):
28     s=0
29     for i in range(J):
30         s+=v[i]**2+v[i+J]**2
31     return s*dx
32
33 L_2=[]
34
35
36 """Moment"""
37 def M(v):
38     s=0
39     dxv=np.dot(mat3,v)
40     for i in range(J):
41         s+=(v[i]-i_comp*v[i+J])*(dxv[i]+i_comp*dxv[i+J])
42     return (s*dx).imag
43
44 Mom=[]
45
46
47 """Variables"""
48 alpha=0
49 beta=0
50 a=3./2.
51 b=-3./5.
52 c=1./10.
53
54 c_u=2
55 v_u=0.5
56 sigma=1
57
58 u=lambda x,t: (((c_u**2-2*c_u*v_u)/4)*(sigma+1))/np.cosh(sigma*np.sqrt((c_u**2-2*
    c_u*v_u)/4)*(x-c_u*t)%L-L/2)**2)*(1/(2*sigma))*np.exp(i_comp*(c_u*(x-v_u*t))
    /2)
59
60 A=np.diag(np.ones(J))+np.diag(np.ones(J-1)*alpha,-1)+np.diag(np.ones(J-1)*alpha,1)
    +np.diag(np.ones(J-2)*beta,-2)+np.diag(np.ones(J-2)*beta,2)
61 A[0][J-1]=alpha
62 A[0][J-2]=beta
63 A[1][J-1]=beta
64 A[J-1][0]=alpha
65 A[J-1][1]=beta
66 A[J-2][0]=beta
67
68 B=np.diag(np.ones(J-1)*(-a/(2*dx)),-1)+np.diag(np.ones(J-1)*(a/(2*dx)),1)+np.diag(
    np.ones(J-2)*(-b/(4*dx)),-2)+np.diag(np.ones(J-2)*(b/(4*dx)),2)+np.diag(np.
    ones(J-3)*(-c/(6*dx)),-3)+np.diag(np.ones(J-3)*(c/(6*dx)),3)
69 B[0][J-1]=-a/(2*dx)
70 B[0][J-2]=-b/(4*dx)
71 B[0][J-3]=-c/(6*dx)
72 B[1][J-1]=-b/(4*dx)
73 B[1][J-2]=-c/(6*dx)
74 B[2][J-1]=-c/(6*dx)
75 B[J-1][0]=a/(2*dx)
76 B[J-1][1]=b/(4*dx)
77 B[J-1][2]=c/(6*dx)
78 B[J-2][0]=b/(4*dx)
79 B[J-2][1]=c/(6*dx)
80 B[J-3][0]=c/(6*dx)

```

```

81
82 A_inv_B=np.matmul(np.linalg.inv(A),B)
83 A_inv_B2=np.matmul(A_inv_B,A_inv_B)
84
85 mat_echange=np.diag(np.ones(J),J)+np.diag(np.ones(J),-J) #premiere matrice de F_n
86
87 mat_somme=np.concatenate((np.eye(J),np.eye(J)),axis=1) #matrice dans la somme de
    F_n
88
89 mat=np.diag(np.ones(2*J))          #matrice possedant des -1 sur la moitiee de sa
    diagonale
90 for i in range(J):
91     mat[i+J][i+J]=-1
92
93 mat_opc2=np.zeros((2*J,2*J)) #matrice avec l'operateur compact au carre
94 for i in range(J):
95     for j in range(J):
96         mat_opc2[i][j]=A_inv_B2[i][j]
97         mat_opc2[i+J][j+J]=A_inv_B2[i][j]
98
99 mat_opc=np.zeros((2*J,2*J))
100 for i in range(J):
101     for j in range(J):
102         mat_opc[i][j]=A_inv_B[i][j]
103         mat_opc[i+J][j+J]=A_inv_B[i][j]
104
105
106 """Fonctions auxiliaires"""
107 def somme(sigma,x,xn): #somme dans F_n
108     s=0
109     for k in range(sigma+1):
110         s+=(np.dot(mat_somme,x*x))**(sigma-k)*(np.dot(mat_somme,xn*xn))**k
111     return s
112
113
114 def tableau_complex(T1,T2): #Fonction auxiliaire creant le tableau T=T1+iT2 terme
    par terme
115     n=np.size(T1)
116     T=np.zeros(J+1,dtype=complex)
117     for i in range(n):
118         T[i]=np.complex(T1[i],T2[i])
119     return T
120
121
122 def F(x,xn): #application tq F(v^(n+1))=0
123     return np.dot(mat,dt/2.*np.dot(np.matmul(mat_opc2,mat_echange),x+xn)+dt/(2.*
        sigma+2.)*np.dot(mat_echange,x+xn)*np.concatenate((somme(sigma,x,xn),somme(
        sigma,x,xn)),axis=0))+(x-xn)
124
125 def Poly(x,xn): #P
126     P=np.zeros(J)
127     for j in range(J):
128         s=0
129         for k in range(sigma+1):
130             s+=(x[j]**2+x[j+J]**2)**(sigma-k)*(xn[j]**2+xn[j+J]**2)**k
131         P[j]=s
132     return P
133
134 def da_P(x,xn): #derivee partielle de P suivant a
135     da_P=np.zeros(J)
136     for j in range(J):
137         s=0

```



```

138         for k in range(sigma):
139             s+=((x[j]**2+x[j+J]**2)**(sigma-k-1))*((xn[j]**2+xn[j+J]**2)**k)*2*x[j
140             ]*(sigma-k)
141             da_P[j]=s
142         return da_P
143
144 def db_P(x,xn): #derivee partielle de P suivant b
145     db_P=np.zeros(J)
146     for j in range(J):
147         s=0
148         for k in range(sigma):
149             s+=((x[j]**2+x[j+J]**2)**(sigma-k-1))*((xn[j]**2+xn[j+J]**2)**k)*2*x[j
150             +J]*(sigma-k)
151             db_P[j]=s
152         return db_P
153
154 def Jac(x,xn): #Jacobiennne de F
155     Jac=np.zeros((2*J,2*J))
156     M_Jac=np.zeros((J,J))
157     N_Jac=np.zeros((J,J))
158     E_Jac=np.zeros((J,J))
159     G_Jac=np.zeros((J,J))
160     for i in range(J):
161         M_Jac[i][i]=(xn[i+J]+x[i+J])*da_P(x,xn)[i]
162         N_Jac[i][i]=Poly(x,xn)[i]+(x[i+J]+xn[i+J])*db_P(x,xn)[i]
163         E_Jac[i][i]=Poly(x,xn)[i]+(x[i]+xn[i])*da_P(x,xn)[i]
164         G_Jac[i][i]=(xn[i]+x[i])*db_P(x,xn)[i]
165     Jac[:,J:J]=np.eye(J)+dt/(2*sigma+2)*M_Jac
166     Jac[:,J:J]=dt/2*A_inv_B2+dt/(2*sigma+2)*N_Jac
167     Jac[J:J,:J]=-dt/2*A_inv_B2-dt/(2*sigma+2)*E_Jac
168     Jac[J:J,:J]=np.eye(J)-dt/(2*sigma+2)*G_Jac
169     return Jac
170
171 """Schema"""
172 V_trace=np.zeros((N+1,J+1),dtype=complex)
173 eps=10**(-12)
174 v0=u(X[:J],0)
175 v=v0
176 va=v.real #separation des parties imaginaires et r elle
177 vb=v.imag
178 va0_graphe=np.zeros(J+1)
179 vb0_graphe=np.zeros(J+1)
180 va0_graphe[:J]=va
181 va0_graphe[J]=va[0] #Condition periodique
182 vb0_graphe[:J]=vb
183 vb0_graphe[J]=vb[0] #Condition periodique
184 Va=[va0_graphe]
185 Vb=[vb0_graphe]
186 V_trace[0,:]=tableau_complex(va0_graphe,vb0_graphe)
187 L_2.append(Norme_L(np.concatenate((va,vb),axis=0)))
188 Hamil.append(H(np.concatenate((va,vb),axis=0)))
189 Mom.append(M(np.concatenate((va,vb),axis=0)))
190
191 for n in range(1,np.size(temps)):
192     xn=np.zeros(2*J)
193     xn[:J]=Va[-1][:J]
194     xn[J:J]=Vb[-1][:J]
195     v=Newton(xn,F,Jac,eps)
196     va_graphe=np.zeros(J+1)
197     vb_graphe=np.zeros(J+1)

```

```

198     va_graphe[:,J]=v[:,J]
199     vb_graphe[:,J]=v[J:]
200     va_graphe[J]=va_graphe[0] #Condition periodique
201     vb_graphe[J]=vb_graphe[0] #Condition periodique
202     Va.append(va_graphe)
203     Vb.append(vb_graphe)
204     V_trace[n,:]=tableau_complex(va_graphe,vb_graphe)
205     Hamil.append(H(v))
206     L_2.append(Norme_L(v))
207     Mom.append(M(v))
208
209
210
211     """Animation"""
212     fig=plt.figure(1)
213     plt.clf()
214     ax=fig.gca()
215     line1, =ax.plot(X, np.abs(u(X,0)), 'r')
216     line2, =ax.plot(X, np.abs(V_trace[:,0]), 'b--')
217     ax.set_title('Temps initial')
218     plt.legend(['sol exacte','sol approchee'], loc='best')
219
220
221     plt.figure(2)
222     plt.plot(temps,np.array(Hamil))
223     plt.plot(temps,np.array(L_2))
224     plt.plot(temps,np.array(Mom))
225
226
227     """Mise      jour de chaque plot"""
228     def runanimate1(n):
229         line1.set_data(X, np.abs(u(X,dt*n)))
230         ax.set_title('Temps t=' + str('{:.3f}'.format(n*dt)))
231
232     def runanimate2(n):
233         line2.set_data(X, np.abs(V_trace[:,n]))
234         ax.set_title('Temps t=' + str('{:.3f}'.format(n*dt)))
235
236
237     ani1=anim.FuncAnimation(fig,runanimate1,frames=np.arange(N+1),interval=1,repeat=
        False)
238     ani2=anim.FuncAnimation(fig,runanimate2,frames=np.arange(N+1),interval=1,repeat=
        False)
239
240     plt.show()

```

Listing 9 – Schéma compact pour l'équation de NLS

## Références

- [1] Mascaret — Wikipedia, l'encyclopédie libre, 2021.
- [2] Méthode de Newton — Wikipedia, l'encyclopédie libre, 2021.
- [3] A. Aftalion, J. Dalibard, and C. Josserand. Équation de Schrödinger non linéaire : des condensats de Bose Einstein aux supersolides. 2010.
- [4] B. Badreddine. *Modélisation de la propagation des solitons dans un milieu non linéaire*. PhD thesis, Université Badji Mokhtar Annaba, 2016.
- [5] S. Benzoni. Master 2 : EDP dispersives, 2009.
- [6] C. Courtès. Cours de Calcul Scientifique, Licence 3 Magistère, Université de Strasbourg, 2020.
- [7] A. de Bouard. Equations dispersives non linéaires. 2005.
- [8] M. Delfour, M. Fortin, and G. Payre. Finite-difference solutions of a non-linear Schrödinger equation. *Journal of computational physics*, 44 :277–288, 1981.
- [9] L. DiMenza. Numerical computation of solitons for optical systems. *ESAIM : Mathematical Modelling and Numerical Analysis*, 43 :173—208, 2009.
- [10] P. Gravel. Solitaires et singuliers solitons. *LEDEVOIR*, 2015.
- [11] H. Kanazawa, T. Matsuo, and T. Yaguchi. A conservative compact finite difference scheme for the KdV equation. *Japan Society for Industrial and Applied Mathematics*, 4 :5–8, 2012.
- [12] S. LeCoz. *Existence et stabilité de solitons, multi-solitons et solutions explosives dans quelques équations dispersives non linéaires*. PhD thesis, Université Paul Sabatier Toulouse, 2018.
- [13] M. Peyrard and T. Dauxois. Physique des solitons. 2007.