Fast accurate computation of the fully nonlinear solitary surface gravity waves

Didier Clamond [a,*], Denys Dutykh [b,c]

[a] Laboratoire J.-A. Dieudonné, Université de Nice – Sophia Antipolis, Parc Valrose, 06108 Nice Cedex 2, France
[b] University College Dublin, School of Mathematical Sciences, Belfield, Dublin 4, Ireland
[c] LAMA, UMR 5127 CNRS, Université de Savoie, Campus Scientifique, 73376 Le Bourget-du-Lac Cedex, France

ARTICLE INFO

ABSTRACT

In this short note, we present an easy to implement and fast algorithm for the computation of the steady solitary gravity wave solution of the free surface Euler equations in irrotational motion. First, the problem is reformulated in a fixed domain using the conformal mapping technique. Second, the problem is reduced to a single equation for the free surface. Third, this equation is solved using Petviashvili's iterations together with pseudo-spectral discretisation. This method has a super-linear complexity, since the most demanding operations can be performed using a FFT algorithm. Moreover, when this algorithm is combined with the multi-precision floating point computations, the results can be obtained to any arbitrary accuracy.

© 2013 Elsevier Ltd. All rights reserved.

1. Introduction

Solitary waves play a central role in nonlinear sciences. They appear in various fields ranging from plasmas physics to hydrodynamics and nonlinear optics [10,22]. In some special cases, solitary wave solutions can be found analytically. For example, explicit expressions are known for integrable models such as KdV and NLS equations, but also for some non-integrable equations such as [4,19]. However, no closed form solutions are known for the practically very important case of the free surface Euler equations. In order to construct these solutions, one has to apply some approximation methods. Historically, asymptotic approximations have been proposed first [6] but, these series being divergent [8], the high-order approximations are only valid in the limit $a \to 0$ (a the wave amplitude). In order to avoid all these limitations, various numerical approaches have been also proposed, e.g. [7,15]. We can mention the boundary integral equation method applied to the computation of solitary waves, see [5,9,13,16] and the references therein.

One of the most widely used methods nowadays is Tanaka's algorithm [20]. In this note, we propose a simpler and faster method based on Petviashvili's iterations [18] leading to a very efficient numerical scheme for the computation of solitary gravity waves of the full Euler equations in the water of finite depth. Petviashvili's

iterations have been successfully used for various simplified water wave equations but, to our knowledge, never for the fully nonlinear problem. The algorithm proposed here is efficient and it allows a very straightforward compact implementation under MATLAB, for example. Such a script can be freely downloaded [1].

The note is organised as follows. In Section 2, we introduce the hypothesis, the notations and the Babenko equation which is equivalent to the original problem. In Section 3, we describe a very simple and fast algorithm to compute the solution, and we give some numerical results. Finally, the main conclusions are outlined in Section 4.

2. Mathematical model

We consider a steady two-dimensional potential flow due to a solitary gravity wave propagating at the surface of a homogeneous fluid layer of constant depth d. Let be (x,y) a Cartesian coordinate system moving with the wave, x being the horizontal coordinate and y the upward vertical one. Since solitary waves are localised in space, the surface elevation tends to zero as $x \to \pm\infty$, and $x = 0$ is the abscissa of the crest. The equations of the bottom, of the free surface and of the mean water level are given correspondingly by $y = -d$, $y = \eta(x)$ and $y = 0$, respectively.

2.1. Governing equation

Let be ϕ, ψ, u and v the velocity potential, the stream function, the horizontal and vertical velocities, respectively, such that $u = \phi_x = \psi_y$ and $v = \phi_y = -\psi_x$. It is convenient to introduce the

* Corresponding author. Tel.: +33 492 076 206; fax: +33 493 517 974.
E-mail addresses: diderc@unice.fr (D. Clamond), Denys.Dutykh@ucd.ie (D. Dutykh).
URLs: http://math.unice.fr/~didierc/ (D. Clamond), http://www.denys-dutykh.com/ (D. Dutykh).

complex potential $f \equiv \phi + i\psi$ (with $i^2 = -1$) and the complex velocity $w \equiv u - iv$ that are holomorphic functions of $z \equiv x + iy$ (i.e., $w = \mathrm{d}f/\mathrm{d}z$). The complex conjugate is denoted with a star (e.g., $z^* = x - iy$), while subscripts 'b' denote the quantities written at the seabed — e.g., $z_b(x) = x - id$, $\phi(x) = \phi(x, y = -d)$ — and subscripts 's' denote the quantities written at the surface — e.g., $z_s(x) = x + i\eta(x)$, $\phi_s(x) = \phi(x, y = \eta(x))$. We emphasise that ψ_s and ψ_b are constants because the surface and the bottom are streamlines.

The far field velocity is such that $(u, v) \rightarrow (-c, 0)$ as $x \rightarrow \pm\infty$, so c is the wave phase velocity observed in the frame of reference where the fluid is at rest at infinity ($c > 0$ if the wave travels to the increasing x-direction). Note that $c = (\psi_b - \psi_s)/d$ due to the mass conservation [12].

The dynamic condition can be expressed in term of the Bernoulli equation

$$2p + 2gy + u^2 + v^2 = c^2, \tag{2.1}$$

where p is the pressure divided by the density and $g > 0$ is the acceleration due to gravity. At the free surface the pressure equals that of the atmosphere which is constant and set to zero without loss of generality, i.e., $p_s = 0$.

2.2. Conformal mapping

Consider the change of independent variable $z \mapsto \zeta \equiv (i\psi_s - f)/c$, that conformally maps the fluid domain $\{-\infty \leqslant x \leqslant \infty; -d \leqslant y \leqslant \eta\}$ into the strip $\{-\infty \leqslant \alpha \leqslant \infty; -d \leqslant \beta \leqslant 0\}$ where $\alpha \equiv \mathrm{Re}(\zeta)$ and $\beta \equiv \mathrm{Im}(\zeta)$ (c.f. Fig. 1). The conformal mapping yields the Cauchy–Riemann relations $x_\alpha = y_\beta$ and $x_\beta = -y_\alpha$, while the complex velocity and the velocity components are

$$w = -c\left(\frac{\mathrm{d}z}{\mathrm{d}\zeta}\right)^{-1}, \quad u = \frac{-cx_\alpha}{x_\alpha^2 + y_\alpha^2}, \quad v = \frac{-cy_\alpha}{x_\alpha^2 + y_\alpha^2}, \quad u^2 + v^2 = \frac{c^2}{x_\alpha^2 + y_\alpha^2}.$$

Introducing new dependent variables $X(\alpha, \beta) \equiv x - \alpha$ and $Y(\alpha, \beta) \equiv y - \beta$, the Cauchy–Riemann relations $X_\alpha = Y_\beta$ and $X_\beta = -Y_\alpha$ hold, while the bottom ($\beta = -d$) and the free surface ($\beta = 0$) impermeabilities yield $Y_b(\alpha) \equiv Y(\alpha, -d) = 0$ and $Y_s(\alpha) \equiv Y(\alpha, 0) = \eta$. The functions X and Y can be expressed in term of X_b solely using some pseudo-differential operators [3]

$$X(\alpha, \beta) = \cos[(\beta + d)\partial_\alpha]X_b(\alpha), \quad Y(\alpha, \beta) = \sin[(\beta + d)\partial_\alpha]X_b(\alpha), \tag{2.2}$$

so that the Cauchy–Riemann relations and the bottom impermeability are fulfilled identically. At the free surface $\beta = 0$, (2.2) yields $X_s(\alpha) = \cos[d\partial_\alpha]X_b(\alpha)$, that can be inverted as $X_b(\alpha) = \sec[d\partial_\alpha]X_s(\alpha)$, and hence the relation (2.2) yields

$$Y_s(\alpha) = \eta(\alpha) = \tan[d\partial_\alpha]X_s(\alpha), \tag{2.3}$$

which relates quantities written at the free surface only. The relation (2.3) can be trivially inverted giving, in particular, $(\partial_\alpha X)_s = \mathcal{C}\{Y_s\} \equiv \partial_\alpha \cot[d\partial_\alpha]Y_s$, where $\mathcal{C}$ is a self-adjoint pseudo-differential operator acting on a pure frequency as

$$\mathcal{C}\{e^{ik\alpha}\} = \begin{cases} k\coth(kd)e^{ik\alpha} & (k \neq 0) \\ 1/d & (k = 0) \end{cases}. \tag{2.4}$$
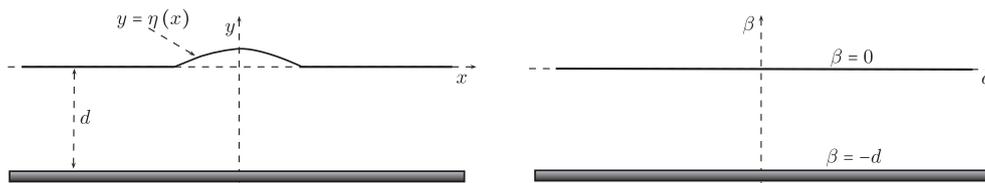
This operator can be efficiently evaluated in the Fourier space using a FFT algorithm.

2.3. Babenko's equation

A Lagrangian for water waves can be obtained considering the kinetic energy minus the potential one, leading, in the conformal plane, to the functional $\mathcal{L} = \int_{-\infty}^{\infty} \mathfrak{L} \, \mathrm{d}\alpha$ with $\mathfrak{L}$ being the Lagrangian density [15]

$$\mathfrak{L} = \frac{1}{2}c^2\eta\mathcal{C}\{\eta\} - \frac{1}{2}g\eta^2(1 + \mathcal{C}\{\eta\}).$$

The governing equation is the Euler–Lagrange equation of this functional, i.e.

$$0 = \frac{\partial \mathfrak{L}}{\partial \eta} + \mathcal{C}\left\{\frac{\partial \mathfrak{L}}{\partial \mathcal{C}\{\eta\}}\right\} = c^2 \mathcal{C}\{\eta\} - g\eta - \frac{1}{2}g\mathcal{C}\{\eta^2\} - g\eta\mathcal{C}\{\eta\}, \tag{2.5}$$

which is the Babenko equation for gravity solitary surface waves [2]. Applying the operator $\mathcal{C}^{-1}$ to (2.5) and splitting the linear and nonlinear parts, the equation can be presented as

$$\mathcal{L}\{\eta\} = \mathcal{N}\{\eta\}, \quad \mathcal{L}\{\eta\} \equiv c^2\eta - g\,\mathcal{C}^{-1}\{\eta\},$$
$$\mathcal{N}\{\eta\} \equiv g\mathcal{C}^{-1}\{\eta\,\mathcal{C}\{\eta\}\} + \frac{1}{2}g\eta^2, \tag{2.6}$$

that is more convenient for the simple and efficient numerical resolution explained below.

3. Numerical method and results

First, we define the wave by choosing its Froude number $F \equiv c/\sqrt{gd}$. Second, we choose a KdV-like initial guess of the free surface elevation $\eta_0(\alpha)$

$$\eta_0(\alpha) \equiv d(F^2 - 1)\,\mathrm{sech}^2(\kappa\alpha/2), \quad (a) \quad F^2 = \tan(\kappa d)\kappa d \quad (b). \tag{3.1}$$

Third, Eq. (3.1-b) is used to compute κ and thence to determine the length L of the computational domain (i.e. $-L \leqslant \alpha \leqslant L$) such that $\eta_0(L) = 0$ to machine precision. The computational domain is then discretised with N equally spaced nodes, with N large enough to avoid aliasing. Fourth, Eq. (2.6) is solved via Petviashvili's iterations [11,17,18]

$$\eta_{n+1} = S_n^2 \mathcal{L}^{-1}\{\mathcal{N}\{\eta_n\}\}, \quad S_n = \frac{\int_{-\infty}^{\infty} \eta_n \mathcal{L}\{\eta_n\} \, \mathrm{d}\alpha}{\int_{-\infty}^{\infty} \eta_n \mathcal{N}\{\eta_n\} \, \mathrm{d}\alpha}, \tag{3.2}$$

where S_n is the co-called *stabilisation factor* which can be computed either in physical or Fourier spaces, thanks to the Parseval identity [21]. In the implementation, we systematically privilege the Fourier space since the operators $\mathcal{C}$ and $\mathcal{C}^{-1}$ can be very efficiently computed according to the definition (2.4) using any FFT algorithm.

The convergence of the iterative process (3.2) is checked by following the norm of the difference between two successive iterations along with the residual in the ℓ_∞ norm

$$\|\eta_{n+1} - \eta_n\|_\infty < \varepsilon_1, \quad \|\mathcal{L}\{\eta_n\} - \mathcal{N}\{\eta_n\}\|_\infty < \varepsilon_2,$$

where ε_n are some prescribed tolerance parameters that are usually of the order of the floating point arithmetics precision.
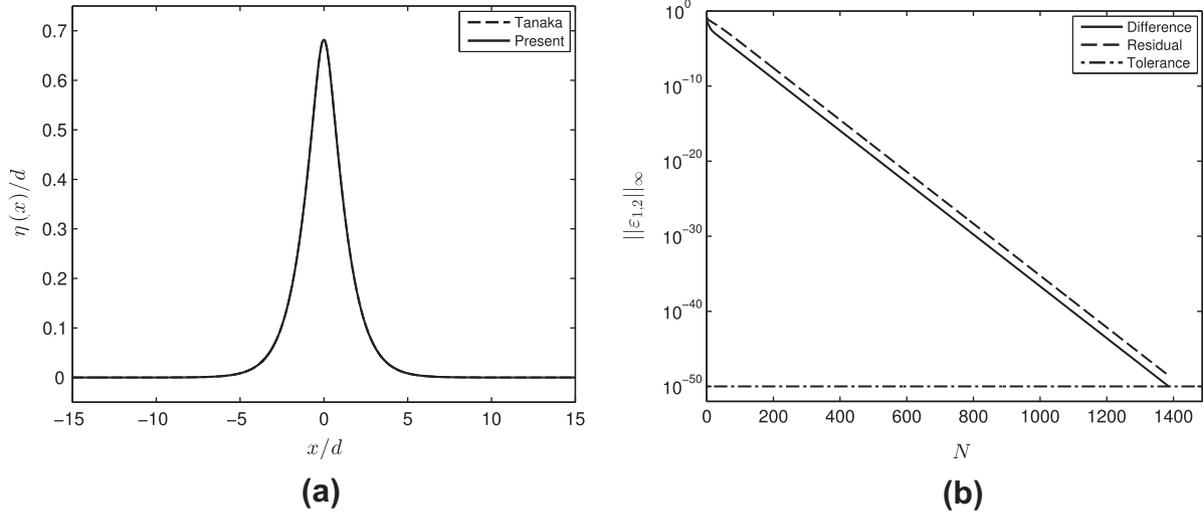

Fig. 1. Sketch of physical and transformed domains.

Fig. 2. Solitary wave profile (a) and convergence of the numerical error during the iterations (b).

Table 1
Amplitude corresponding to $F = 1.274236297977903$ (wrong digits are underlined).

Tanaka (16 digits)	Present (16 digits)	Present (50 digits)
0.6819<u>448200954614</u>	0.68196566738690<u>21</u>	0.681965667386915720709857779975770373038440370273<u>43</u>

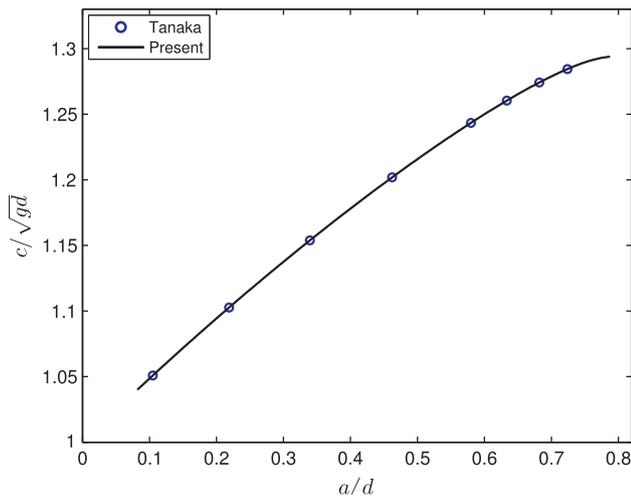**Fig. 3.** Speed versus amplitude relation predicted by the present (solid line) and Tanaka's (circles) algorithms.
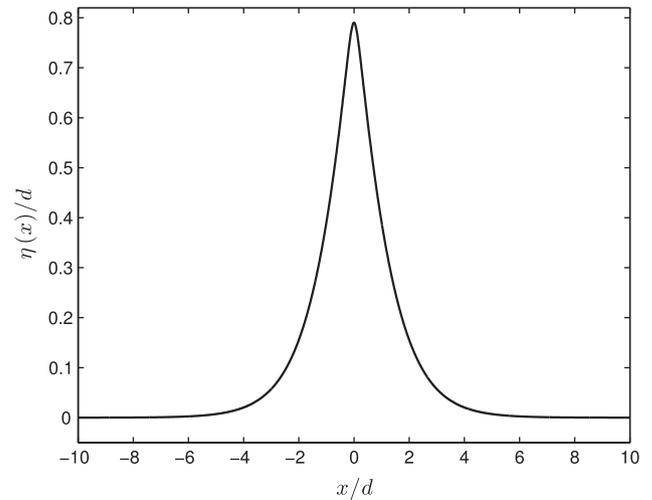
3.1. Numerical results

The present algorithm is illustrated on the following example of a fairly high amplitude solitary wave. It is validated and compared against Tanaka's algorithm [20]. Tanaka's solution is parametrized by the dimensionless parameter $q_c = |u_c|/\sqrt{gd} \in (0, 1)$, where u_c is the fluid velocity at the wave crest. We choose the special value $q_c = 0.4$ for comparisons. Tanaka's algorithm was implemented in MATLAB[1] in its original version without any peculiar optimisations. The interval $-30 \leqslant \phi \leqslant 30$ is discretised using 20,001 nodes and

---

[1] In the sequel, all the algorithms are compared in the same computing environment: MATLAB ®.

Fig. 4. A very high amplitude solitary wave ($a/d \approx 0.790$).

the iterations are continued until the tolerance 10^{-10} is obtained between two successive values of the Froude number F, i.e.

$$\left| F_{n+1}^2 - F_n^2 \right| < \varepsilon_3 (= 10^{-10} \text{ in our computation}).$$

This computation required 47 iterations which lasted 385 s on our desktop computer. The resulting Froude number computed by Tanaka's algorithm is

$$F = 1.274236297977903.$$

Then, we take this Froude number and use it as the solitary wave parameter in the Babenko Eq. (2.6). We use a (periodised) domain of the same length as in Tanaka's method, and discretise it with $N = 16,384$ nodes. The tolerance is chosen to be 10^{-15}, i.e. very close to machine accuracy. These parameters are kept in all

the computations presented below. The iterative procedure is stopped when the ℓ_∞ norm of the difference between two successive iterations is less than the tolerance (the norm of the residual is checked a posteriori). Our algorithm required 372 iterations to converge and the computations lasted for about 0.5 s, i.e. almost three orders of magnitude faster than Tanaka's algorithm. This flagrant difference in CPU times can be explained by the fact that our method has the super-linear complexity $\emptyset(N\log N)$ while Tanaka's method has a quadratic complexity $\mathcal{O}(N^2)$.

Moreover, the present algorithm can be used to compute solitary waves to an arbitrary precision. To illustrate this, we implemented the scheme using a Multi-Precision Computing Toolbox (MPCT) for MATLAB [14]. The code modifications needed to implement the arbitrary accuracy are minor; this constitutes one of the major advantages of this MPCT. In these higher-precision experiments, all floating point operations are done with 50 significant digits. The tolerance parameter is set to 10^{-50}. To achieve such extreme accuracy, the Petviashvili scheme required 1386 iterations. The convergence of the algorithm is shown on Fig. 2b. One can notice that the residual error is slightly higher than the difference between two successive iterations, likely due to the presence of quadratic nonlinearities into the equation. However, both measures of the numerical error decrease monotonically with iterations. The resulting solitary wave profile is depicted on Fig. 2a along with the corresponding Tanaka solution. The two curves cannot be distinguished up to the graphical resolution, thus validating our method. The respective accuracies can be asserted comparing the dimensionless amplitudes a/d (Table 1).

One can notice that despite the tolerance parameter set to 10^{-10} in Tanaka's algorithm, its real accuracy is seriously degraded for this large amplitude wave (only five digits of the wave amplitude are predicted correctly). Conversely, the present method gives fourteen correct digits with a tolerance of 10^{-15} in double precision calculation (about sixteen digits), and 49 correct digits using the 50-digit computation (the latter was determined using a 60-digit calculation with a tolerance of 10^{-55} with eight times more nodes over an interval twice the size).

The algorithm being validated, it can be used to produce some physically sound numerical results. For example, we compute to high-precision (here 10^{-15}, but it is not a limitation) the speed–amplitude relation for solitary waves. The result is shown on Fig. 3. We also computed several values in a wide range of amplitudes using Tanaka's algorithm (represented with circles). Again, up to the graphical resolution we have a very good agreement between the two methods. However, the speed of Tanaka's algorithm does not allow to compute many values in a reasonable time.

We note finally that, for simplicity, we defined the wave by its Froude number which varies in the interval $1 \leqslant F \leqslant 1.29421$. Doing so, we can compute all waves such that $0 \leqslant a \lesssim 0.796$ that is sufficient for all cases of practical interest. One of the largest solitary waves which can be predicted by this parametrization is represented on Fig. 4. However, the solutions of larger amplitudes exist, the limit $a \lesssim 0.796$ being obtained here because F is not a monotonic function of a (i.e., $F(a)$ is multi-valued when a is large). Thus, in order to compute larger waves, another parameter (e.g., a/d or q_c) should be used to define the wave. This requires simple modifications in the code [1]. We successfully did it and were able to compute all waves such that $a \lesssim 0.82$. For even higher waves, the present method is not efficient because these extreme waves being (almost) singular at the crest, the Fourier series does not converge fast enough and another spectral decomposition should be used instead.

4. Conclusions

We have described a fast, accurate and easy to implement method for computing steady gravity solitary surface waves solution of the full Euler equations. The method is based on first rewriting the problem into the equivalent but simpler Babenko's equation and, second, using Petviashvili's iterations to solve it numerically. The resulting scheme allows the computation to any arbitrary accuracy using a multi-precision floating-point arithmetics library [14]. All ingredients of the method are well known, but their combination seems to be new in this context and turns out to be very efficient. Our MATLAB implementation is rather compact: The computational numerical core is about a dozen lines of code, the rest being pre- and post-processing. The script [1] can be freely downloaded and used (it can also be easily translated into any programming language).

Acknowledgements

References

[1] http://www.mathworks.com/matlabcentral/fileexchange/39189-solitary-water-wave. 2, 7, 8.
[2] Babenko KI. Some remarks on the theory of surface waves of finite amplitude. Sov Math Dokl 1987;35:599–603.
[3] Clamond D. Steady finite-amplitude waves on a horizontal seabed of arbitrary depth. J Fluid Mech 1999;398:45–60.
[4] Clamond D, Dutykh D. Practical use of variational principles for modeling water waves. Physica D 2012;241(1):25–36.
[5] Evans WAB, Ford MJ. An exact integral equation for solitary waves (with new numerical results for some 'internal' properties). Proc R Soc Lond A 1996;452:373–90.
[6] Fenton J. A ninth-order solution for the solitary wave. J Fluid Mech 1972;53(2):257–71.
[7] Fenton JD, Rienecker MM. A Fourier method for solving nonlinear water-wave problems: application to solitary-wave interactions. J Fluid Mech 1982;118:411–43.
[8] Germain J-P. Contribution à l'étude de la houle en eau peu profonde. Thèse de l'Université de Grenoble, France; 1967 [in French].
[9] Hunter J, Vanden-Broeck J-M. Solitary and periodic gravity-capillary waves of finite amplitude. J Fluid Mech 1983;134:205–19.
[10] Infeld E, Rowlands G. Nonlinear waves, solitons and chaos. Cambridge University Press; 2000.
[11] Lakoba TI, Yang J. A generalized Petviashvili iteration method for scalar and vector Hamiltonian equations with arbitrary form of nonlinearity. J Comput Phys 2007;226:1668–92.
[12] Longuet-Higgins MS. On the mass, momentum, energy and circulation of a solitary wave. Proc R Soc Lond A 1974;337:1–13.
[13] Maklakov DV. Almost-highest gravity waves on water of finite depth. Eur J Appl Math 2002;13:67–93.
[14] Multiprecision computing toolbox for MATLAB. v3.3.8.2611. Advanpix LLC., Tokyo, Japan; 2012.
[15] Okamoto H, Shōji M. The mathematical theory of permanent progressive water-waves. Advanced series in nonlinear dynamics. World Scientific Pub Co Inc.; 2002.
[16] Parau E, Vanden-Broeck J-M, Cooker M. Nonlinear three-dimensional gravity-capillary solitary waves. J Fluid Mech 2005;536:99–105.
[17] Pelinovsky D, Stepanyants YA. Convergence of Petviashvili's iteration method for numerical approximation of stationary solutions of nonlinear wave equations. SIAM J Numer Anal 2004;42:1110–27.
[18] Petviashvili VI. Equation of an extraordinary soliton. Sov J Plasma Phys 1976;2(3):469–72.
[19] Serre F. Contribution à l'étude des écoulements permanents et variables dans les canaux. La Houille blanche 1953;8:374–872 [in French].
[20] Tanaka M. The stability of solitary waves. Phys Fluids 1986;29(3):650–5.
[21] Titchmarsh EC. The theory of functions. 2nd ed. Oxford University Press; 1976.
[22] Yang J. Nonlinear waves in integrable and non-integrable systems. SIAM; 2010.