Algorithmic Computation of Generalized Symmetries of Nonlinear Evolution and Lattice Equations *

ÜNAL GÖKTAŞ † AND WILLY HEREMAN

Department of Mathematical and Computer Sciences,
Colorado School of Mines, Golden, CO 80401-1887, U.S.A.
E-mail: umalg@wolfram.com, whereman@mines.edu.

A straightforward algorithm for the symbolic computation of generalized (higher-order) symmetries of nonlinear evolution equations and lattice equations is presented. The scaling properties of the evolution or lattice equations are used to determine the polynomial form of the generalized symmetries. The coefficients of the symmetry can be found by solving a linear system. The method applies to polynomial systems of PDEs of first-order in time and arbitrary order in one space variable. Likewise, lattices must be of first order in time but may involve arbitrary shifts in the discretized space variable.

The algorithm is implemented in Mathematica and can be used to test the integrability of both nonlinear evolution equations and semi-discrete lattice equations. With our Integrability Package, generalized symmetries are obtained for several well-known systems of evolution and lattice equations. For PDEs and lattices with parameters, the code allows one to determine the conditions on these parameters so that a sequence of generalized symmetries exists. The existence of a sequence of such symmetries is a predictor for integrability.

Subject classification: AMS(MOS) 35Q53, 35R10, 34K99, 68Q40.
Keywords: Symmetry, Integrability, Evolution Equation, Lattice, PDE, DDE

1 Introduction

A large number of physically important nonlinear models are completely integrable, which means that they are linearizable via an explicit transformation or solvable by the Inverse Scattering Transform. Integrable continuous or discrete models arise in key branches of physics including classical, quantum, particle, statistical, and plasma physics. Integrable equations also model wave phenomena in nonlinear optics and the bio-sciences. Mathematically, nonlinear models involve

*Research supported in part by the NSF under Grant CCR-9625421.
†Current address: Wolfram Research, Inc., 100 Trade Center Drive, Champaign, IL 61820-7237, U.S.A.
ordinary or partial differential equations (ODEs or PDEs), differential-difference equations (DDEs), integral equations, etc. [15].

Whichever form they come in, completely integrable equations exhibit analytic properties reflecting their rich mathematical structure. For instance, completely integrable PDEs and DDEs possess infinitely many symmetries and conserved quantities (if the model is conservative). Perhaps after a suitable change of variables, the equations have the Painlevé property, admit Bäcklund transformations, prolongation structures, or can be written in bi-Hamiltonian form [15].

The existence of an infinite hierarchy of symmetries for integrable equations can be established by explicitly constructing the recursion operator that connects such symmetries. Finding symmetries and recursion operators for nonlinear models is a nontrivial task, in particular if attempted with pen and paper. Computer algebra systems can greatly assist in the search for generalized symmetries and recursion operators. See e.g. [20, 28] for an algorithm to find recursion operators.

In this paper we present a direct algorithm [20] that allows one to automatically compute polynomial generalized (higher-order) symmetries for polynomial PDEs in 1+1 dimensions and polynomial DDEs (semi-discrete lattices). The systems of DDEs or PDEs must be of evolution type, i.e. first order in (continuous) time. The number of equations, the degree of nonlinearity, and the order of differentiation (or shift level) in the spatial variable are arbitrary.

We use the dilation invariance of the given system of PDEs or DDEs to determine the form of the polynomial generalized symmetry. Upon substitution of the form of the symmetry into the defining equation, one has to solve a linear system for the unknown constant coefficients of the symmetry. In case the original system contains free parameters, the eliminant of that linear system will determine the necessary conditions for the parameters, so that the system admits the required generalized symmetry. Our algorithm can thus be used as an integrability test for classes of equations involving parameters.

For the PDE case, a slight extension of our algorithm allows one to compute generalized symmetries that explicitly depend on the independent variables \(x\) (space) and \(t\) (time). However, in such cases, it is necessary to specify the highest degree of the independent variables in the generalized symmetry.

Once the generalized symmetries are explicitly known, it is quite often possible to find the recursion operator by inspection [14]. If the recursion operator is hereditary then the equation will possess infinitely many symmetries. If the operator is hereditary and factorizable then the equation has infinitely many conserved quantities [15, 18].

With respect to nonlinear DDEs, a comprehensive integrability study was done by Yamilov and co-workers (see e.g. [49, 55]). Using the formal symmetry approach [11, 38, 50], they were able to give a classification of semi-discrete equations possessing infinitely many local conservation laws and symmetries, and
provide an algorithm to construct them.

For Lagrangian systems the set of higher-order symmetries can be shown to lead to the set of conservation laws [31, 42, 43, 45]. For equations without Lagrangian structure there is no universal correspondence between symmetries and conservation laws. Recent results on this subject are given in [5] for ODEs, and in [4, 6] for PDEs. For Mathematica algorithms and code to compute conservation laws of nonlinear PDEs and DDEs we refer to [21] and [22, 23, 24]. The relationship between symmetries and conservation laws, as expressed through Noether’s theorem, is beyond the scope of this paper (see [45] for details).

The algorithm presented in this paper has been implemented in Mathematica and it can be used to test the integrability of nonlinear systems of PDEs and DDEs, provided they are of polynomial type. With our Integrability Package, we computed polynomial generalized symmetries for several well-known systems of evolution and lattice equations [20]. As some examples in Sections 4 and 5 show, for PDEs and DDEs with parameters the code automatically determines the conditions on these parameters so that a sequence of generalized symmetries exists.

The existence of a sequence of higher-order symmetries is a predictor for integrability. Our program is a tool for the search of the first few higher-order (or generalized) symmetries. An existence proof (showing that there are indeed infinitely many higher-order symmetries) must be done analytically, e.g. by explicitly constructing the recursion operator [43] that connects these symmetries.

If our program finds as many generalized symmetries as the number of components in the given PDE or DDE system, then there is a good chance that the system has infinitely many higher-order symmetries and that the recursion operator can be constructed explicitly. If one cannot find a sufficient number of polynomial higher-order symmetries (see Bakirov’s system in Section 4.2) then it is unlikely that the PDE or DDE system is completely integrable, at least in that coordinate representation.

In essence, our software does not allow one to conclude that an equation is not integrable because polynomial generalized symmetries could not be found. Polynomial PDEs or DDEs that lack polynomial higher-order symmetries may accidentally have other types of symmetries. It is also feasible that PDEs or DDEs without polynomial symmetries could be transformed into equations that do have polynomial symmetries.

The paper is organized as follows. In Section 2 we give the definition of generalized symmetry. We also give the three steps of the algorithm to compute generalized symmetries of nonlinear evolution equations. The Korteweg-de Vries and Boussinesq equations are used to illustrate the computations. In Section 3 we extend the algorithm to nonlinear DDEs (the semi-discrete case), with the Toda lattice as leading example. The examples in Section 4 show how the technique
works for complex equations, like the nonlinear Schrödinger equation and one of its discretizations (the Ablowitz-Ladik system), and systems with more than two components, like the Benney system. The Hirota-Satsuma system illustrates the method for equations with one parameter. Section 5 is devoted to a more extensive application involving a parameterized class of fifth-order KdV-type equations. In Section 6 we give details about the use of our symmetry package in Mathematica. We briefly discuss other software for generalized symmetries in Section 7. Conclusions and an outlook for future research are given in Section 8.

2 Generalized Symmetries of Partial Differential Equations

2.1 Definition

Consider a system of PDEs in the (single) space variable \( x \) and time variable \( t \),

\[
\mathbf{u}_t = \mathbf{F}(\mathbf{u}, \mathbf{u}_x, \mathbf{u}_{2x}, \ldots, \mathbf{u}_{mx}),
\]

where \( \mathbf{u} \) and \( \mathbf{F} \) are vector dynamical variables with the same number of components: \( \mathbf{u} = (u_1, u_2, \ldots, u_n) \), \( \mathbf{F} = (F_1, F_2, \ldots, F_n) \) and \( \mathbf{u}_{mx} = \frac{\partial^n \mathbf{u}}{\partial x^n} \). The vector function \( \mathbf{F} \) is assumed to be a polynomial function in \( \mathbf{u}, \mathbf{u}_x, \ldots, \mathbf{u}_{mx} \). There are no restrictions on the order of the system or its degree of nonlinearity. If PDEs are of second or higher order in \( t \), we assume that they can be recast in the form (1).

A vector function \( \mathbf{G}(x, t, \mathbf{u}, \mathbf{u}_x, \mathbf{u}_{2x}, \ldots) \), with \( \mathbf{G} = (G_1, G_2, \ldots, G_m) \), is called a generalized symmetry of (1) if and only if it leaves (1) invariant for the replacement \( \mathbf{u} \rightarrow \mathbf{u} + \epsilon \mathbf{G} \) within order \( \epsilon \). Hence,

\[
\mathbf{D}_t(\mathbf{u} + \epsilon \mathbf{G}) = \mathbf{F}(\mathbf{u} + \epsilon \mathbf{G})
\]

must hold up to order \( \epsilon \) on the solutions of (1). Consequently, \( \mathbf{G} \) must satisfy the linearized equation [13, 15]

\[
\mathbf{D}_t \mathbf{G} = \mathbf{F}'(\mathbf{u})[\mathbf{G}],
\]

where \( \mathbf{F}' \) is the Fréchet derivative of \( \mathbf{F} \), i.e.,

\[
\mathbf{F}'(\mathbf{u})[\mathbf{G}] = \frac{\partial}{\partial \epsilon} \mathbf{F}(\mathbf{u} + \epsilon \mathbf{G})|_{\epsilon=0}.
\]

In (2) and (4) we infer that \( \mathbf{u} \) is replaced by \( \mathbf{u} + \epsilon \mathbf{G} \), and \( \mathbf{u}_{mx} \) by \( \mathbf{u}_{mx} + \epsilon \mathbf{D}_x^m \mathbf{G} \). As usual, \( \mathbf{D}_t \) and \( \mathbf{D}_x \) are total derivatives.

In the case of point symmetries, \( \mathbf{G}(x, t, \mathbf{u}, \mathbf{u}_x, \mathbf{u}_t) \) depends linearly on \( \mathbf{u}_x \) and \( \mathbf{u}_t \). By classical symmetries one means the set of point and contact symmetries, where \( \mathbf{G}(x, t, \mathbf{u}, \mathbf{u}_x, \mathbf{u}_t) \). All other symmetries, including higher derivatives than the first, are called generalized symmetries [43]. Other authors call them higher-order or Lie-Bäcklund symmetries [31, 40].
The symmetry $G(x, t, u, u_x, u_{2x}, \ldots)$ is the characteristic of the evolutionary vector field [43, p. 291] $X = \sum_{i=1}^{n} G_i(x, t, u, u_x, u_{2x}, \ldots) \frac{\partial}{\partial u_i}$. This vector field is also called the canonical Lie-Bäcklund operator [31, p. 58].

The examples used in the description of the algorithm, involve one or two dependent variables. For simplicity of notation, the components of $u$ will be denoted by $u, v, \ldots$ (instead of $u_1, u_2$, etc.).

2.2 Algorithm

To illustrate our algorithm, we consider the Korteweg-de Vries (KdV) equation

$$u_t = 6uu_x + u_{3x}, \quad (5)$$

from soliton theory [1, 42]. This ubiquitous evolution equation is long known to have infinitely many symmetries [14].

Key to our method is the observation that (5) is invariant under the dilation symmetry (or scaling)

$$(t, x, u) \rightarrow (\lambda^{-3} t, \lambda^{-1} x, \lambda^2 u), \quad (6)$$

where $\lambda$ is an arbitrary parameter. The result of this dimensional analysis can be stated as follows: $u$ corresponds to two derivatives with respect to $x$, for short, $u \sim \frac{\partial^2}{\partial x^2}$. Similarly, $\frac{\partial}{\partial t} \sim \frac{\partial}{\partial x^3}$. Scaling invariance, which is a special Lie-point symmetry, is an intrinsic property of many integrable nonlinear PDEs and DDEs.

Our algorithm exploits this scaling invariance to find generalized symmetries, which now proceeds in three steps.

Step 1: Determine the weights of variables

The weight, $w$, of a variable is by definition equal to the number of derivatives with respect to $x$ the variable carries. Weights are rational, and weights of dependent variables are nonnegative. We set $w(\frac{\partial}{\partial x}) = 1$. In view of (6), we have $w(u) = 2$ and $w(\frac{\partial}{\partial t}) = 3$. Consequently, $w(x) = -1$ and $w(t) = -3$.

The rank of a monomial is defined as the total weight of the monomial, again in terms of derivatives with respect to $x$. Observe that (5) is an equation of rank 5, since all the terms (monomials) have the same rank, namely 5. This property is called uniformity in rank.

Conversely, requiring uniformity in rank for (5) allows one to compute the weights of the dependent variables. Indeed, with $w(\frac{\partial}{\partial x}) = 1$ we have

$$w(u) + w(\frac{\partial}{\partial t}) = 2w(u) + 1 = w(u) + 3,$$

which yields $w(u) = 2$, $w(\frac{\partial}{\partial t}) = 3$. Hence, $w(t) = -3$, which is consistent with (6).
Step 2: Construct the form of the symmetry

As an example, let us compute the form of the symmetry of rank 7. Start by listing all powers in \( u \) with rank 7 or less: \( \mathcal{L} = \{1, u, u^2, u^3\} \). Next, for each monomial in \( \mathcal{L} \), introduce enough \( x \)-derivatives, so that each term exactly has rank 7. Thus,

\[
\frac{\partial}{\partial x}(u^3) = 3u^2u_x, \quad \frac{\partial^3}{\partial x^3}(u^2) = 6u_xu_{2x} + 2uu_{3x}, \quad \frac{\partial^5}{\partial x^5}(u) = u_{5x}, \quad \frac{\partial^7}{\partial x^7}(1) = 0.
\]

Then, gather the resulting (non-zero) terms in a set \( \mathcal{R} = \{ u^2u_x, uu_{2x}, uu_{3x}, u_{5x}\} \), which contains the building blocks of the symmetry. Linear combination of the monomials in \( \mathcal{R} \) with constant coefficients \( c_i \) gives the form of the symmetry:

\[
G = c_1u^2u_x + c_2uu_{2x} + c_3uu_{3x} + c_4u_{5x}. \tag{7}
\]

Step 3: Determine the unknown coefficients in the symmetry

We determine the coefficients \( c_i \) by requiring that (3) holds on the solutions of (1). Compute \( D_tG \) and use (1) to remove \( u_t, u_{tx}, u_{txx}, \text{etc.} \). For given \( F \), compute the Fréchet derivative (4) and, in view of (3), equate the resulting expressions. Treating the different monomial terms in \( u \) and its \( x \)-derivatives as independent, the linear system for the coefficients \( c_i \) is readily obtained.

For (5), we perform this computation with \( F = 6uu_x + u_{3x} \) and \( G \) in (7). Considering as independent all products and powers of \( u, u_x, u_{xx}, \ldots \), in

\[
(12c_1-18c_2)u^2_xu_{2x} + (6c_1-18c_3)uu_{2x}^2 + (6c_1-18c_3)uu_{x3} + (3c_2-60c_1)u_{3x}^2 + (3c_2+3c_3-90c_4)u_{2x}u_{4x} + (3c_3-30c_4)u_{x5} = 0,
\]

we obtain the linear system for the coefficients \( c_i \):

\[
S = \{12c_1-18c_2 = 0, 6c_1-18c_3 = 0, 3c_2-60c_1 = 0, 3c_2+3c_3-90c_4 = 0, 3c_3-30c_4 = 0\}.
\]

The solution is \( c_1 = \frac{c_2}{3} = \frac{c_3}{5} = c_4 \). Since symmetries can only be determined up to a multiplicative constant, we choose \( c_1 = 30, c_2 = 20, c_3 = 10 \) and \( c_4 = 1 \), and substitute this into (7). Hence,

\[
G = 30u^2u_x + 20uu_{2x} + 10uu_{3x} + u_{5x}.
\]

Note that \( u_t = G \) is known as the Lax equation, which is the fifth-order PDE in the completely integrable KdV hierarchy [37].

Analogously, for (5) we computed the \((x-t)\) independent symmetries of rank \( \leq 11 \). They are:

\[
G^{(1)} = u_x, \quad G^{(2)} = 6uu_x + u_{3x}, \quad G^{(3)} = 30u^2u_x + 20uu_{2x} + 10uu_{3x} + u_{5x},
\]

\[
G^{(4)} = 140u^3u_x + 70u^3 + 280uu_xu_{2x} + 70u^2u_{3x} + 70uu_xu_{3x} + 42u_{x4x}
+ 14uu_{5x} + u_{7x},
\]
\[ G^{(5)} = 630u_4^4u_x + 1260uu_3^3u_x + 2520u_2^2u_xu_2u_x + 1302u_xu_2^2u_x + 420u_3^3u_x + 966u_2^2u_3x + 1260uu_2u_3x + 756uu_xu_4 + 252u_3xu_4 + 126u_3u_5x + 168u_2u_5x + 72u_xu_6x + 18uu_7x + u_{9x}. \]

These results agree with those listed in the literature (see e.g. [14, 40, 43]).

**Remarks.**

(i) The recursion operator [43, p. 312] for the KdV equation is given by
\[ \mathcal{R} = D^2 + 4u + 2u_xD^{-1}. \] (9)

This operator is hereditary [18] and connects the above symmetries. For example,
\[ \mathcal{R}u_x = (D^2 + 4u + 2u_xD^{-1})u_x = 6uu_x + u_{3x}, \]
\[ \mathcal{R}(6uu_x + u_{3x}) = (D^2 + 4u + 2u_xD^{-1})(6uu_x + u_{3x}) = 30u^2u_x + 20u_2u_x + 10u_3u_x + u_{5x}, \] (10)

and so forth.

Note that the recursion operator (9) is also uniform in rank with respect to the dilation symmetry (6). In [20, 28] we give a simple algorithm to construct such operators.

(ii) Instead of working with the definition (3) of the symmetry, one could introduce an evolution equation,
\[ u_x = G(x, t, u, u_x, u_{2x}, ...), \] (11)
which defines the flow generated by \( G \) and parameterized by the auxiliary time variable \( \tau \). The symmetry can then be computed from the compatibility condition of (1) and (11):
\[ D_\tau F(u, u_x, u_{2x}, ..., u_{mx}) = D_\tau G(x, t, u, u_x, u_{2x}, ..., u_{mx}). \] (12)

One then proceeds as follows: As above, determine the form of the symmetry \( G \) involving the constant coefficients \( c_i \). Then, compute \( D_\tau G \) and use (1) to remove \( u_t, u_{x}, \) etc. Subsequently, compute \( D_\tau F \) and use (11) to remove \( u_t, u_{xx}, \) etc. Finally, use (12) to determine the linear system for the unknown \( c_i \). Solve the system and substitute the result into the form of \( G \).

Applied to our example, \( D_\tau G \) is computed with \( G \) in (7). Next, (5) is used to eliminate all \( t \)-derivatives of \( u \) from the expression of \( D_\tau G \). Then, we computed \( D_x F \) with \( F \) in the right hand side of (5), and eliminated all \( \tau \)-derivatives through (11) after substitution of (7). Finally, expressing that \( D_x F - D_\tau G \equiv 0 \) we again obtained (8).

Although this procedure (see Ito [33]) circumvents the evaluation of the Fréchet derivative, it seems more involved than our algorithm which uses the definition (3).
2.3 Symmetries Explicitly Dependent on $x$ and $t$

The KdV equation (5) has also symmetries which explicitly depend on $x$ and $t$. Our algorithm can be used to find these symmetries provided that we specify the maximum degree in $x$ and $t$.

As an example, we compute the symmetry of rank 2 for (5), that linearly depends on $x$ and/or $t$. In other words, the highest degree in $x$ or $t$ in the symmetry is 1.

We start with the list of monomials in $u, x$ and $t$ of rank 2 or less:

$$\mathcal{L} = \{1, u, x, xu, t, tu, tu^2\}.$$  

Then, for each monomial in $\mathcal{L}$, introduce enough $x$-derivatives, so that each term exactly has weight 2. Thus,

$$D_x(xu) = u + xu_x, \quad D_x(tu^2) = 2tuu_x, \quad D_x^2(tu) = t u^3x, \quad D_x^2(1) = D_x^2(t) = 0.$$  

Gather the non-zero resulting terms in a set $\mathcal{R} = \{u, xu_x, tuu_x, tuu_x^3\}$, which contains the building blocks of the symmetry. Linear combination of the monomials in $\mathcal{R}$ with constant coefficients $c_i$ gives the form of the symmetry:

$$G = c_1 u + c_2 xu_x + c_3 tuu_x + c_4 tuu_x^3.$$  

(13)

Now, determine the coefficients $c_1$ through $c_4$ by requiring that (3) holds on the solutions of (5). After grouping the terms, one gets

$$(6c_1 + 6c_2 - c_3)u u_x + (3c_3 - 18c_4)tu^2u_x + (3c_2 - c_4)u_3x + (3c_3 - 18c_4)tu_xu_3x \equiv 0,$$

which yields

$$\mathcal{S} = \{6c_1 + 6c_2 - c_3 = 0, 3c_3 - 18c_4 = 0, 3c_2 - c_4 = 0\}.$$  

The solution is $\frac{2}{3} = 3c_2 = \frac{2}{3} = c_1$. We choose $c_1 = \frac{2}{3}, c_2 = \frac{2}{3}, c_3 = 6$ and $c_4 = 1$, and substitute this into (13). Hence,

$$G = \frac{2}{3} u + \frac{1}{3} xu_x + 6tuu_x + tuu_x^3.$$  

Similarly, we computed other symmetries of (5) that linearly depend on $x$ and $t$. They are of rank 0 and 2:

$$G^{(1)} = 1 + 6tu_x, \quad \text{and} \quad G^{(2)} = 2u + xu_x + 3tu_t = 2u + xu_x + 18tuu_x + 3tuu_x^3.$$  

(14)

These two symmetries correspond to the Lie-point symmetries of (5), with generators $X = t \frac{\partial}{\partial x} - \frac{\partial}{\partial t}$ (Galilean boost) and $X = 3t \frac{\partial}{\partial x} + x \frac{\partial}{\partial x} - 2u \frac{\partial}{\partial t}$ (scaling), respectively [43, p. 291]. Note that with $R$ in (9), $\mathcal{R}G^{(1)} = G^{(2)}$, but $\mathcal{R}G^{(2)}$ is no longer a ‘local’ symmetry (it involves integrals). The symmetries in (14) agree with those in e.g. [40].
2.4 Example: The Boussinesq Equation

For scaling invariant systems such as (5), it suffices to consider the dilation symmetry on the space of independent and dependent variables. For systems that are inhomogeneous under a suitable scaling symmetry, such as the example given below, we use the following trick: We introduce one (or more) auxiliary parameter(s) with an appropriate scaling. These extra parameters can be viewed as additional dependent variables, however, their derivatives are zero. By extending the action of the dilation symmetry to the space of independent and dependent variables, including the parameters, we are able to apply our algorithm to a larger class of polynomial PDE systems.

Consider the wave equation,

\[ u_{tt} - u_{2x} + 3uu_{2x} + 3u_x^2 + \alpha u_{4x} = 0, \]  

(\(\alpha\) constant) which was proposed by Boussinesq to describe surface water waves whose horizontal scale is much larger than the depth of the water [1].

To apply our algorithm, we must first rewrite (15) as a first-order system,

\[ u_t = v_x, \quad v_t = u_x - 3uu_x - \alpha u_{3x}, \]  

where \(v\) is an auxiliary dependent variable. It is easy to verify that the terms \(u_x\) and \(\alpha u_{3x}\) in the second equation obstruct uniformity in rank. To circumvent the problem we introduce an auxiliary parameter \(\beta\) with (unknown) weight, and replace (16) by

\[ u_t = v_x, \quad v_t = \beta u_x - 3uu_x - \alpha u_{3x}. \]  

As described in Step 1 we compute the weights from

\[ w(u) + w(\frac{\partial}{\partial t}) = w(v) + 1, \]
\[ w(v) + w(\frac{\partial}{\partial t}) = w(\beta) + w(u) + 1 = 2w(u) + 1 = w(u) + 3. \]

This yields

\[ w(u) = 2, \quad w(v) = 3, \quad w(\beta) = 2, \quad \text{and} \quad w(\frac{\partial}{\partial t}) = -w(t) = 2, \]

and the scaling properties of (17) are \(u \sim \beta \sim \frac{\partial}{\partial t} \sim \frac{\partial^2}{\partial x^2}, \quad v \sim \frac{\partial}{\partial x}. \) Indeed, (17) is invariant under the dilation symmetry

\[ (t, x, u, v, \beta) \rightarrow (\lambda^{-2}t, \lambda^{-1}x, \lambda^2 u, \lambda^3 v, \lambda^2 \beta). \]

Observe that all the monomials in the equations in (17) have rank 4 and 5. Therefore, for any symmetry \(G\) of (17),

\[ \text{rank}(G_2) = \text{rank}(G_1) + 1 = \text{rank}(G_1) + w(v) - w(u). \]
Let us construct the form of the symmetry $\mathbf{G} = (G_1, G_2)$ with rank$(G_1) = 6$ and rank$(G_2) = 7$. First, list all monomials in $u, v$ and $\beta$ of rank 6 (respectively rank 7) or less:

$$\mathcal{L}_1 = \{1, \beta, \beta^2, u, \beta u, \beta^2 u, u^2, \beta u^2, u^2 v, v, \beta v, u v, v^2\},$$

$$\mathcal{L}_2 = \{1, \beta, \beta^2, u, \beta u, \beta^2 u, u^2, \beta u^2, u^3, v, \beta v, \beta^2 v, u v, \beta u v, u^2 v, v^2\}.$$

Next, for each monomial in $\mathcal{L}_1$ and $\mathcal{L}_2$, introduce the necessary $x$-derivatives, so that each term in $\mathcal{L}_1$ exactly has rank 6, and each term in $\mathcal{L}_2$ has rank 7. Keeping in mind that $\beta$ is constant, and proceeding with the rest of the algorithm, we obtain:

$$G_1^{(1)} = u x v + u v_x + \frac{2}{3} \alpha w_3 x,$$

$$G_2^{(1)} = \beta u u_x - 3 u^2 u_x + v v_x - 6 \alpha u_x u_{2x} + \frac{2}{3} \alpha \beta u_{3x} - 3 \alpha u u_{3x} - \frac{2}{3} \alpha^2 u_{5x}.$$

Finally, setting $\beta = 1$ in (18), one obtains a symmetry of (16) although initially this system was not uniform in rank, and neither are these symmetries. We list one more generalized symmetry of (16):

$$G_1^{(2)} = u u_x - \frac{3}{2} u^2 u_x + v v_x - 5 \alpha u_x u_{2x} + \frac{2}{3} \alpha u_{3x} - 2 \alpha u u_{3x} - \frac{8}{15} \alpha^2 u_{5x},$$

$$G_2^{(2)} = u v_x + u v_x - 3 \alpha u v_x - \frac{3}{2} u^2 v_x - 2 \alpha u_{2x} v_x - 3 \alpha u_x u_{2x} - \alpha u_{3x} v_x + \frac{2}{3} \alpha w_{3x}$$

$$- 2 \alpha u u_{3x} - \frac{8}{15} \alpha^2 v_{5x}.$$

### 3 Generalized Symmetries of Differential-difference Equations

#### 3.1 Definition

Consider a system of DDEs,

$$\dot{u}_n = \mathbf{F}(..., u_{n-1}, u_n, u_{n+1}, ...),$$

where the equations are continuous in time, and discretized in the (single) space variable. As before, $u_n$ and $\mathbf{F}$ are vector dynamical variables with any number of components, and $\mathbf{F}$ is assumed to be a polynomial with constant coefficients. There are no restrictions on the level of shifts or the degree of nonlinearity. If DDEs are of second or higher order in $t$, they must be recast in the form (19).

A vector function $\mathbf{G}(..., u_{n-1}, u_n, u_{n+1}, ...)$ is called a generalized symmetry of (19) if the infinitesimal transformation

$$u_n \rightarrow u_n + \epsilon \mathbf{G}(..., u_{n-1}, u_n, u_{n+1}, ...)$$

### (20)
leaves (19) invariant within order $\epsilon$.

Consequently, $G$ must satisfy the linearized equation [13, 15]

\[ D_\epsilon G = F'(u_n)[G], \tag{21} \]

where $F'$ is the Fréchet derivative of $F$, defined as

\[ F'(u_n)[G] = \frac{\partial}{\partial \epsilon} F(u_n + \epsilon G)|_{\epsilon=0}. \tag{22} \]

Of course, (20) means that $u_{n+k}$ is replaced by $u_{n+k} + \epsilon G|_{n \to n+k}$. For compactness of notation, in (21) and (22) we used $F'(u_n)$ instead of $F'(\ldots, u_{n-1}, u_n, u_{n+1}, \ldots)$.

Also for notational simplicity, in the description of the algorithm below, the components of $u_n$ will be denoted by $u_n, v_n$, etc. We use $F_1, F_2, \ldots$ and $G_1, G_2, \ldots$ to denote the components of $F$ and $G$, respectively.

### 3.2 Algorithm

As the leading example, we consider the one-dimensional lattice [25, 53]

\[ \dot{y}_n = \exp(y_{n-1} - y_n) - \exp(y_n - y_{n+1}), \tag{23} \]

due to Toda. In (23), $y_n$ is the displacement from equilibrium of the $n$th particle with unit mass under an exponential decaying interaction force between nearest neighbors. With the change of variables,

\[ u_n = \dot{y}_n, \quad v_n = \exp(y_n - y_{n+1}), \]

the Toda lattice (23) can be written in polynomial form

\[ \dot{u}_n = v_{n-1} - v_n, \quad \dot{v}_n = v_n(u_n - u_{n+1}). \tag{24} \]

Observe that (24) is invariant under the dilation symmetry

\[ (t, u_n, v_n) \rightarrow (\lambda^{-1} t, \lambda u_n, \lambda^2 v_n), \tag{25} \]

where $\lambda$ is an arbitrary parameter. Thus, $u_n$ corresponds to one derivative with respect to $t$, or $u_n \sim \frac{d}{dt}$, and, similarly, $v_n \sim \frac{d^2}{dt^2}$.

Our 3-step algorithm for symmetries exploits only this type of scaling transformation. Note that we do not consider any scaling of the discrete variable $n$.

**Step 1:** Determine the weights of variables

In contrast to the algorithm for PDEs, we have to define the weight, $w$, of variables in terms of the number of derivatives with respect to $t$, and we set $w\left(\frac{d}{dt}\right) = 1$. Weights of dependent variables are nonnegative, rational, and independent of $n$. In view of (25), we have $w(u_n) = 1$, and $w(v_n) = 2$. 
The *rank* of a monomial is defined as the total weight of the monomial, again in terms of derivatives with respect to \( t \). Observe that in the first equation of (24), all the monomials have the same rank, namely 2, and in the second equation, all the monomials have rank 3.

Conversely, requiring uniformity in rank for each equation in (24) allows one to compute the weights of the dependent variables. Indeed,

\[
\begin{align*}
  w(u_n) + 1 &= w(v_n), \\
  w(v_n) + 1 &= w(u_n) + w(v_n),
\end{align*}
\]

yields \( w(u_n) = 1, w(v_n) = 2 \), which is consistent with (25).

*Step 2: Construct the form of the symmetry*

As an example, we compute the form of the symmetry of rank \((3, 4)\), i.e. \( G_1 \) and \( G_2 \) will have ranks 3 and 4, respectively. Start by listing all monomials in \( u_n \) and \( v_n \) of ranks 3 and 4, or less:

\[
\begin{align*}
  \mathcal{L}_1 &= \{ u_n^3, u_n^2, u_n v_n, u_n, v_n \}, \\
  \mathcal{L}_2 &= \{ u_n^4, u_n^3, u_n^2 v_n, u_n^2, u_n v_n, u_n, v_n^2, v_n \}.
\end{align*}
\]

Next, for each monomial in \( \mathcal{L}_1 \) and \( \mathcal{L}_2 \), introduce the necessary \( t \)-derivatives, so that each term exactly has rank 3 and 4, respectively. At the same time, use (24) to remove all \( t \)-derivatives. Doing so, based on \( \mathcal{L}_1 \), we obtain

\[
\begin{align*}
  \frac{d^0}{dt^0}(u_n^3) &= u_n^3, \\
  \frac{d^0}{dt^0}(u_n v_n) &= u_n v_n, \\
  \frac{d}{dt}(v_n) &= \dot{v}_n = u_n v_n - u_{n+1} v_n,
\end{align*}
\]

Gather the resulting terms in a set: \( \mathcal{R}_1 = \{ u_n^3, u_n^2 v_n, u_n, v_n, u_n, v_n \} \).

Similarly, based on the monomials in \( \mathcal{L}_2 \), we get

\[
\begin{align*}
  \mathcal{R}_2 &= \{ u_n^4, u_n^3 v_n, u_n^2, u_n^2 v_n, u_n^2, u_n v_n, u_n, v_n^2, v_n \}.
\end{align*}
\]

Linear combination of the monomials in \( \mathcal{R}_1 \) and \( \mathcal{R}_2 \) with constant coefficients \( c_i \) gives the explicit form of the symmetry:

\[
\begin{align*}
  G_1 &= c_1 u_n^3 + c_2 u_n^2 v_n - 1 + u_n v_n = c_3 u_n v_n + c_4 u_n v_n + c_5 u_n v_n, \\
  G_2 &= c_6 u_n^4 + c_7 u_n^3 v_n - 1 + u_n^2 v_n - 1 + c_8 u_n v_{n-1} + c_9 u_n^2 v_{n-1} + c_{10} v_n - 2 v_{n-1} + c_{11} v_n^2 + c_{12} u_n^2 v_n + c_{13} u_n v_{n-1} + c_{14} u_n^2 v_n + c_{15} v_n v_{n-1} + c_{16} v_n^2 + c_{17} v_n v_{n+1}.
\end{align*}
\]

(26)
Step 3: Determine the unknown coefficients in the symmetry

To determine the coefficients $c_i$ we require that (21) holds on any solution of (19). Compute $D_t G$ and use (19) to remove all $\dot{u}_{n-1}, \dot{u}_n, \dot{u}_{n+1}$, etc. Compute the Fréchet derivative (22) and, in view of (21), equate the resulting expressions. Considering as independent all the monomials in $u_n$ and their shifts, we obtain the linear system that determines the coefficients $c_i$.

Applied to (24) with (26), we obtain the solution

$$
c_1 = c_6 = c_7 = c_8 = c_9 = c_{10} = c_{11} = c_{13} = c_{36} = 0, \quad (27)
$$

$$
-c_2 = -c_3 = c_4 = c_5 = -c_{12} = c_{14} = -c_{15} = c_{17}. \quad (28)
$$

Therefore, with the choice $c_{17} = 1$, the symmetry is

$$
G_1 = u_n v_n - u_{n-1} v_{n-1} + u_{n+1} v_n - u_n v_{n-1},
$$

$$
G_2 = u_{n+1} v_n - u_n v_n + v_n v_{n+1} - v_{n-1} v_n. \quad (29)
$$

It is easy to produce new completely integrable DDEs based on these symmetries. For instance, the DDE system

$$
\dot{u}_n = G_1 = u_n v_n - u_{n-1} v_{n-1} + u_{n+1} v_n - u_n v_{n-1},
$$

$$
\dot{v}_n = G_2 = u_{n+1} v_n - u_n v_n + v_n v_{n+1} - v_{n-1} v_n. \quad (30)
$$

is also completely integrable.

To illustrate the effectiveness of our algorithm to ‘filter out’ integrable cases among systems of DDEs with parameters, consider a parameterized version of the Toda lattice,

$$
\dot{u}_n = \alpha v_{n-1} - v_n, \quad \dot{v}_n = v_n (\beta u_n - u_{n+1}), \quad (31)
$$

where $\alpha$ and $\beta$ are nonzero constants. In [44] it was shown that (31) is completely integrable if and only if $\alpha = \beta = 1$.

Using our algorithm, one can easily compute the compatibility conditions for $\alpha$ and $\beta$, so that (31) admits a polynomial symmetry, say, of rank (3, 4). The steps are the same as for (24). However, the linear system for the $c_i$ is parameterized by $\alpha$ and $\beta$ and must be analyzed carefully (with e.g. Gröbner basis methods). This analysis leads to the condition $\alpha = \beta = 1$.

For $\alpha = \beta = 1$, (31) coincides with (24), for which we computed symmetries with ranks (4, 5) and (5, 6). They are:

$$
G_1^{(1)} = u_n v_n + u_n u_{n+1} v_n + u_{n+1} v_n + v_n v_{n+1} - u_{n-1} u_n v_{n-1} - u_{n-1} v_n v_{n-1} - u_{n-2} v_{n-1} - v_{n-2} v_{n-1},
$$

$$
G_2^{(1)} = u_{n+1} v_n + 2 u_{n+1} v_n v_{n+1} + u_n + 2 v_n v_{n+1} - u_n v_n + u_{n+1} v_n - u_n v_{n-1} - 2 u_n v_{n-1} v_n - u_n v_n. \quad (32)
$$
\[ G_1^{(2)} = u_n^3 v_n + u_n^2 u_{n+1} v_n + u_n u_n^2 v_n + u_n^3 v_n + 2 u_n u_n^2 + 2 u_{n+1} u_n^2 
+ u_n v_n v_{n+1} + 2 u_{n+1} v_n v_{n+1} + u_{n+2} v_n v_{n+1} - u_{n-1} v_n v_{n-1} 
- u_{n-1} u_n^2 v_{n-1} - u_n^3 v_{n-1} - u_{n-2} v_n - 2u_{n-1} v_{n-2} v_{n-1} 
- u_n v_n - 2u_{n-1} u_n^2 v_{n-1} - 2u_n v_n^2 v_{n-1}, \]

\[ G_2^{(2)} = u_n^4 v_n - u_n^3 v_n - u_n^2 v_{n-1} v_n - 2 u_{n-1} u_n v_{n-1} v_n - 3u_n v_{n-1} v_n 
- v_{n-2} v_{n-1} v_n - v_{n-1}^2 v_n - 2 u_n^2 v_n^2 + 2 u_{n+1} v_n^2 - v_{n-1}^2 v_n + 3 u_n^2 v_n v_{n+1} 
+ 2 u_{n+1} u_{n+2} v_n v_{n+1} + u_{n+2}^2 v_n v_{n+1} + v_n^2 v_{n+1} + v_n v_{n+1} v_{n+2}. \] (32)

Starting from (24), the change of variables \((t, u_n, v_n) \rightarrow (-t, U_{n-1}, V_n)\) gives the lumped network system [41]:

\[ \dot{U}_n = V_{n+1} - V_n, \quad \dot{V}_n = V_n(U_n - U_{n-1}). \] (33)

The symmetries of (33) follow from (32) under the same change of variables.

4 More Examples of PDEs and DDEs

4.1 Nonlinear Schrödinger Equation

The nonlinear Schrödinger (NLS) equation [1],

\[ i \partial_t q - \partial^2_{xx} q + 2 |q|^2 q = 0, \] (34)

arises as an asymptotic limit of a slowly varying dispersive wave envelope in a nonlinear medium, and as such has significant applications in nonlinear optics, water waves, and plasma physics. Together with the ubiquitous KdV equation (5), the completely integrable NLS equation is one of the most studied soliton equations.

In order to compute the symmetries of (34) we consider \( q \) and \( q^* \) as independent variables and add the complex conjugate equation to (34). Absorbing \( i \) in the scale of \( t \), we get

\[ q_t - q_{2x} + 2q^2 q^* = 0, \quad q_t^* + q_{2x}^* - 2q^2 q = 0. \] (35)

Since \( w(q) = w(q^*) \), we obtain

\[ w(q) = w(q^*) = 1, \quad \text{and} \quad w\left( \frac{\partial}{\partial t} \right) = -w(t) = 2. \]

We computed the symmetries of ranks (4,4), (5,5), and (6,6):

\[ G_1^{(1)} = -6qqx q^* + q_{3x}, \quad G_2^{(1)} = G_1^{(1)} = -6qqx q^* + q_{3x}. \]
\[
\begin{align*}
G_1^{(2)} &= -6q^3q^{*2} + 6d_2q^* + 4q_dq^*_d + 8qq_2q^*_d + 2q^*_d q^*_d - q_4x, \\
G_2^{(2)} &= G_1^{* (2)} \\
G_1^{(3)} &= 30q^2q^*_d q^*_d - 10q^*_d q^*_d - 20q_d q_2 q^*_d - 10qq_2 q^*_d - 10q^*_d q^*_d - 10q^*_d q^*_d - q_5x, \\
G_2^{(3)} &= G_1^{* (3)}. \tag{36}
\end{align*}
\]

4.2 Bakirov’s System

The following system, due to I. M. Bakirov (see [43, p. 381, ex. 5.16(b)]):

\[
u_t = 5(u_{4x} + v^2), \quad v_t = v_{4x}, \tag{37}
\]

is peculiar in the sense that it has exactly one nontrivial higher-order symmetry, as recently proved in [8]. System (37) serves as a counterexample to the long-standing claim that the existence of one higher-order symmetries implies integrability, i.e., the existence of infinitely many higher-order symmetries. However, (37) does not contradict the conjecture by Fokas [15] which says that a system with \( n \) components should have \( n \) symmetries to be integrable.

For (37), \( w_{(2)} = 4 \) and \( w(u) = 2w(v) - 4 \), where \( w(v) \) is still free. We searched for symmetries for two choices: (i) \( w(u) = 0, w(v) = 2 \), and (ii) \( w(u) = 4, w(v) = 4 \). With the latter choice we obtained the symmetries of rank \((4,4), (8,8)\) and \((10,10)\):

\[
\begin{align*}
G_1^{(1)} &= 2u, \quad G_2^{(1)} = v, \\
G_1^{(2)} &= 5v^2 + 5u_{4x}, \quad G_2^{(2)} = v_{4x}, \\
G_1^{(3)} &= 20v_{x}^2 + 25uv_{2x} + 11u_{6x}, \quad G_2^{(3)} = v_{6x}.
\end{align*}
\]

Obviously, \( G^{(1)} \) and \( G^{(2)} \) are trivial, and \( G^{(3)} \) is the sixth-order nontrivial symmetry, also obtained in [8] and [9].

4.3 The Hirota-Satsuma System

Hirota and Satsuma [30] proposed a coupled system of KdV equations,

\[
u_t = 6\alpha uu_x + 2u_{x} + \alpha u_{3x}, \quad v_t = 3uv_x + v_{3x}, \tag{38}
\]

where \( \alpha \) is a nonzero parameter. System (38) describes the interaction of two long waves with different dispersion relations. It is known to be completely integrable provided \( \alpha = -\frac{1}{2} \).

The scaling properties of (38) are such that \( w(u) = w(v) = 2 \) and \( w_{(2)} = 3 \), if we set \( w_{(2)} = 1 \). So, \( u \sim v \sim \frac{\partial^2}{\partial x^2}, \frac{\partial^3}{\partial x^3} \). System (38) has rank 5. A search for symmetries produced this symmetry of rank 7:

\[
\begin{align*}
G_1 &= u^2u_x - 2\frac{2}{3}u_xv^2 - 4uvv_x + 2u_xu_{2x} - \frac{2}{3}v_xv_{2x} + \frac{1}{3}uu_{3x} - \frac{2}{3}v_{3x} + \frac{1}{30}u_{5x}, \\
G_2 &= -\frac{1}{3}u^2v_x - \frac{2}{3}v^2v_x - \frac{1}{3}u_{2x}v_{2x} - 2u_xv_{2x} - \frac{2}{3}u_xv_{2x} - \frac{2}{3}v_{3x} - \frac{2}{15}v_{5x},
\end{align*}
\]
provided that $\alpha = -\frac{1}{2}$, which is the condition for complete integrability of (38).

4.4 Benney’s System

As a model for nonlinear water waves, Benney [7] proposed the three-component system:

$$
\begin{align*}
   u_t &= vv_x + 2u_xw + 2uw_x, \\
   v_t &= 2u_x + v_xw + vw_x, \\
   w_t &= 2v_x + 2uw_x.
\end{align*}
\right.
$$

With $w(\frac{\partial}{\partial x}) = 1$, we have $w(\frac{\partial}{\partial y}) = w(w) + 1, w(u) = 3w(w), w(v) = 2w(w)$, where $w(w)$ is arbitrary. Consequently, (39) is invariant under the family of scaling transformations,

$$
(t, x, u, v, w) \rightarrow (\lambda^{-d-1}t, \lambda^{-1}x, \lambda^{3d}u, \lambda^{2d}v, \lambda^d w),
$$

where $d$ is an arbitrary nonnegative integer. We selected $d = 1$, and computed the symmetries of rank $(4, 3, 2), (5, 4, 3), (6, 5, 4)$ and $(7, 6, 5)$, respectively. Here are the results:

$$
\begin{align*}
   G_1^{(1)} &= u_x, \\
   G_2^{(1)} &= v_x, \\
   G_3^{(1)} &= w_x, \\
   G_1^{(2)} &= (v^2 + 4uw)_x, \\
   G_2^{(2)} &= (4u + 2vw)_x, \\
   G_3^{(2)} &= (4v + 2w^2)_x, \\
   G_1^{(3)} &= (4uw + v^2w + 3uw^2)_x, \\
   G_2^{(3)} &= (4uw + 3v^2 + vw^2)_x, \\
   G_3^{(3)} &= (4u + 4vw + w^3)_x, \\
   G_1^{(4)} &= (12u^2 + 24uw + 8uw^3 + 4v^3 + 3v^2w^2)_x, \\
   G_2^{(4)} &= (24uw + 12uw^2 + 12v^2w + 2vw^3)_x, \\
   G_3^{(4)} &= (24uw + 12v^2 + 12vw^2 + 2w^4)_x.
\end{align*}
$$

4.5 Volterra Lattices

Consider the integrable discretization of the KdV equation:

$$
u_n = u_n (u_{n+1} - u_{n-1}),
$$

which is also known as the Kac–Van Moerbeke equation. It arises in the study of Langmuir oscillations in plasmas and in population dynamics [1, 34].

The lattice (41) is invariant under the dilation symmetry $(t, u_n) \rightarrow (\lambda^{-1}t, \lambda u_n)$. In terms of weights, $w(u_n) = 1$ if $w(\frac{d}{dt}) = 1$. Hence, $u_n$ corresponds to one derivative with respect to $t$, i.e. $u_n \sim \frac{d}{dt}$. 
Note that the more general class of equations
\[ \dot{v}_n = v_n \left( v_{n+1}^p - v_{n-1}^p \right), \] (42)
with \( p \) integer, is related to (41) by the change of variables \( u_n = p w_n^p \). So, it suffices to study (41), for which we computed the symmetries of rank 3 through 5. They are:

\[
\begin{align*}
G^{(1)} &= u_n u_{n+1} (u_n + u_{n+1} + u_{n+2}) - u_{n-1} u_n (u_{n-2} + u_{n-1} + u_n), \\
G^{(2)} &= u_n^3 u_{n+1} + 2 u_n^2 u_{n+1}^2 + u_n u_{n+1}^3 + u_n^2 u_{n+1} u_{n+2} + 2 u_n u_{n+1}^2 u_{n+2} \\
&\quad + u_n u_{n+1} u_{n+2} + u_n u_{n+1} u_{n+2} u_{n+3} - u_{n-3} u_{n-2} u_{n-1} u_n - u_{n-2} u_{n-1} u_n \\
&\quad - 2 u_{n-2} u_{n-1} u_n - u_{n-3} u_{n-2} u_{n-1} u_n - u_{n-2} u_{n-1} u_n - u_{n-1} u_n^3, \\
G^{(3)} &= u_n^4 u_{n+1} + u_{n-1} u_n^3 u_{n+1} + 3 u_n u_{n+1}^2 u_{n+2} + 3 u_n^2 u_{n+1}^3 + u_n u_{n+1}^4 + u_n u_{n+1} u_{n+2} \\
&\quad + 4 u_n^2 u_{n+1} u_{n+2} + 3 u_n^3 u_{n+1} u_{n+2} + u_n^2 u_{n+1}^2 u_{n+2} + 3 u_n u_{n+1}^2 u_{n+2} \\
&\quad + u_n u_{n+1} u_{n+2} + 2 u_n u_{n+1} u_{n+2} + 2 u_n u_{n+1} u_{n+2} + 2 u_n u_{n+1} u_{n+2} \\
&\quad + u_n u_{n+1} u_{n+2} + u_{n-1} u_n u_{n-2} u_{n-1} u_n - u_{n-1} u_n u_{n-2} u_{n-1} u_n \\
&\quad - u_{n-2} u_{n-1} u_n - u_{n-3} u_{n-2} u_{n-1} u_n - u_{n-4} u_{n-3} u_{n-2} u_{n-1} u_n \\
&\quad - u_{n-2} u_{n-1} u_n - u_{n-3} u_{n-2} u_{n-1} u_n - u_{n-4} u_{n-3} u_{n-2} u_{n-1} u_n \\
&\quad - u_{n-2} u_{n-1} u_n - u_{n-3} u_{n-2} u_{n-1} u_n - u_{n-4} u_{n-3} u_{n-2} u_{n-1} u_n \\
&\quad - u_{n-2} u_{n-1} u_n - u_{n-3} u_{n-2} u_{n-1} u_n - u_{n-4} u_{n-3} u_{n-2} u_{n-1} u_n.
\end{align*}
\]

Ignoring a trivial misprint in [40], Mikhailov et al. listed the symmetry \( G^{(1)} \).

4.6 The Ablowitz-Ladik Discretization of the Nonlinear Schrödinger Equation

In [2, 3], Ablowitz and Ladik studied some of the properties of the following integrable discretization of the NLS equation:
\[ i \dot{u}_n = u_{n+1} - 2u_n + u_{n-1} \pm u_n^* u_n (u_{n+1} + u_{n-1}), \] (43)
where \( u_n^* \) is the complex conjugate of \( u_n \). We continue with the plus sign; the other case would be treated similarly. Instead of splitting \( u_n \) into real and imaginary parts, we treat \( u_n \) and \( v_n = u_n^* \) as independent variables and augment (43) with its complex conjugate equation. Absorbing \( i \) in the scale on \( t \), we get
\[
\begin{align*}
\dot{u}_n &= u_{n+1} - 2u_n + u_{n-1} + u_n v_n (u_{n+1} + u_{n-1}), \\
\dot{v}_n &= - (v_{n+1} - 2v_n + v_{n-1}) - u_n v_n (v_{n+1} + v_{n-1}).
\end{align*}
\] (44)

Since \( v_n = u_n^* \), we have \( w(v_n) = w(u_n) \). Neither of the equations in (44) is uniform in rank. To circumvent this problem we introduce an auxiliary parameter \( \alpha \) with weight, and replace (44) by
\[
\begin{align*}
\dot{u}_n &= \alpha (u_{n+1} - 2u_n + u_{n-1}) + u_n v_n (u_{n+1} + u_{n-1}), \\
\dot{v}_n &= -\alpha (v_{n+1} - 2v_n + v_{n-1}) - u_n v_n (v_{n+1} + v_{n-1}).
\end{align*}
\] (45)
Uniformity in rank requires that
\[
\begin{align*}
    w(u_n) + 1 &= w(\alpha) + w(u_n) = 2w(u_n) + w(v_n) = 3w(u_n), \\
    w(v_n) + 1 &= w(\alpha) + w(v_n) = 2w(v_n) + w(u_n) = 3w(v_n),
\end{align*}
\]
which yields \( w(u_n) = w(v_n) = \frac{1}{2}, w(\alpha) = 1 \), or, \( u_n^2 \sim v_n^2 \sim \alpha \sim \frac{d}{dt} \).

Recall that the ‘uniformity in rank’ requirement is essential for the first two steps of the algorithm. However, after step 2, we may set \( \alpha = 1 \). The computations now proceed as in the previous examples. We searched for symmetries of (44) of ranks \((2,2)\) through \((7/2,7/2)\), and found symmetries of ranks \((5/2,5/2)\) and \((7/2,7/2)\). To save space, we only list the symmetries of rank \((5/2,5/2)\):

\[
\begin{align*}
    G_1^{[1]} &= -u_{n+2} - u_n u_{n+1} v_{n-1} - u_n^2 v_{n-1} - u_n^3 u_{n+2} v_{n-1} - u_n^2 u_{n+1} v_n - u_n^3 u_{n+3} v_{n-1}, \\
    G_2^{[1]} &= v_{n+2} + u_n u_{n-1} v_{n-1} + u_n v_{n-2} v_{n-1} + u_n v_{n-1} v_{n+1} + u_n^2 v_{n-1}, \\
    G_1^{[2]} &= -u_{n-2} - u_n u_{n-1} v_{n-1} - u_n^2 v_{n-1} - u_n^3 u_{n-1} v_{n-1} - u_n^2 u_{n+1} v_n - u_n^3 u_{n+3} v_{n-1}, \\
    G_2^{[2]} &= u_n v_{n+2} u_{n+1} + u_n u_{n-1} v_{n+1} + u_n v_{n-1} v_{n+1} + u_n^2 v_{n+1} + u_n^3 v_{n+2} + v_{n+2} + u_n^3 v_{n+1} v_{n+1} + u_n u_{n+1} v_n u_{n+1} v_{n+1}.
\end{align*}
\]

4.7 Generalized Toda Lattices

In [51], the integrability of the chain
\[
\dot{y}_n = v_{n+1} e^{y_{n+1} - y_n} - e^{2(y_{n+1} - y_n)} - y_{n-1} e^{y_n - y_{n-1}} + e^{2(y_n - y_{n-1})},
\]
which is related to the relativistic Toda lattice has been studied. With the change of variables, \( u_n = \dot{y}_n, \ v_n = \exp(y_{n+1} - y_n) \), lattice (46) can be written as
\[
\begin{align*}
    \dot{u}_n &= v_n (u_n + u_{n-1} - v_n - u_{n-1} (u_n - v_n)), \\
    \dot{v}_n &= u_n (u_{n+1} - u_n).
\end{align*}
\]
Here, \( u_n \sim v_n \sim \frac{d}{dt} \). We computed a couple of symmetries for (47). One of them reads:

\[
\begin{align*}
    G_1 &= u_n^2 v_{n-1} + u_n u_{n-1} v_{n-1} + u_n v_{n-2} - v_n^2 v_{n-1} - 2 u_n v_{n-2} - 2 u_{n-1} v_n^2 \\
    &\quad - u_n^2 v_{n-1} + v_{n-1} - u_n u_{n+1} v_{n-1} + u_n v_{n-1} v_{n+1} + u_n v_{n+1} - u_n v_{n+1}^2 \\
    &\quad - u_n^2 - u_n v_{n+1} v_{n+1} + v_{n+1} v_{n+1}, \\
    G_2 &= u_n v_n - u_n^2 v_{n+1} + u_n u_{n-1} v_{n-1} - v_{n-1} v_{n-1} - u_n v_n^2 + u_n v_{n+1}^2 \\
    &\quad - u_n^2 v_{n+1} + v_{n+1} v_{n+1}.
\end{align*}
\]
In [52], Suris investigated the integrability of
\[ \ddot{y}_n = \dot{y}_n \left[ \exp \left( y_{n+1} - y_n \right) - \exp \left( y_n - y_{n-1} \right) \right], \] (48)
which is closely related to the classical Toda lattice (23). The same change of variables as for (46) allows one to rewrite (48) as
\[ \dot{u}_n = u_n (v_n - v_{n-1}), \quad \dot{v}_n = v_n (u_{n+1} - u_n). \] (49)
Again, \( u_n \sim v_n \sim \frac{\partial}{\partial n} \). We computed three symmetries. Two of them are:

\begin{align*}
G_1^{(1)} &= u_n^2 v_n + u_n u_{n+1} v_n + u_n u_{n-1}^2 v_{n-1} - u_{n-1} u_n v_{n-1} - u_n v_{n-1}^2, \\
G_2^{(1)} &= u_{n+1}^2 v_n + u_{n+1} u_{n+2} v_{n+1} - u_{n+1}^2 v_{n+1} - u_n v_{n+1} - u_{n+1} v_{n+1}^2, \\
G_1^{(2)} &= u_n^2 v_n + u_n u_{n+1} v_n + u_n u_{n-1}^2 v_{n-1} + 2 u_n^2 v_n + 2 u_{n+1}^2 v_{n+1} + u_n^3 v_n \\
&\quad + u_n u_{n+1} v_{n+1} - u_{n-1}^2 u_n v_{n-1} - u_n u_{n-1}^2 v_{n-1} - u_n^2 v_{n-1}, \\
&\quad - u_n u_{n+1} v_{n+1}^2 v_{n+1} - 2 u_{n-1}^2 u_n v_{n-1} - u_n^2 u_{n-1}^2 - u_n^3 v_{n-1}, \\
G_2^{(2)} &= u_n^3 v_n - u_n^3 v_n - u_n u_{n+1} v_{n+1} - u_n v_{n+2} v_{n+1} - u_{n-1}^2 v_{n+1} - u_n v_{n+1} - 2 u_{n+2}^2 v_n \\
&\quad + 2 u_{n+1}^2 v_n - u_n v_{n+1}^2 - u_n v_{n+1}^2 + u_n^2 v_{n+1}^2 + 2 u_{n+1} v_{n+1} v_{n+1} + u_{n+1} v_{n+1} v_{n+1}. \\
\end{align*}
(50)

Note that the change of variables \( (u_n, v_n) \to (U_{n-1}, V_n) \) in (49) gives the Volterra system [41]:
\[ \dot{U}_n = U_n (V_{n+1} - V_n), \quad \dot{V}_n = V_n (U_n - U_{n-1}). \] (51)
The symmetries of (51) follow from (50) under the same transformation.

5 Application

We perform the integrability analysis of a class of fifth-order KdV equations,
\[ u_t = \alpha u_x^2 u_x + \beta u_x u_{2x} + \gamma u u_{3x} + u_{5x}, \] (52)
where \( \alpha, \beta, \gamma \) are nonzero constant parameters.

Integrable cases of (52) are well known in the literature [16, 29, 36, 47]. Indeed, for \( \alpha = 30, \beta = 20, \gamma = 10 \), equation (52) reduces to the Lax equation [37]. The SK equation, due to Sawada and Kotera [48] and Dodd and Gibbon [12], is obtained for \( \alpha = 5, \beta = 5, \gamma = 5 \). The KK equation, due to Kaup [35] and Kupershmidt, corresponds to \( \alpha = 20, \beta = 25, \gamma = 10 \).
The scaling properties of (52) are such that \( w(u) = 2 \) and \( w(\frac{\partial u}{\partial x}) = 5 \), if \( w(\frac{\partial y}{\partial x}) = 1 \). So, \( u \sim \frac{\partial^2 u}{\partial x^2} \) and \( \frac{\partial u}{\partial x} \sim \frac{\partial^3 u}{\partial x^3} \). Using our algorithm, one easily computes the \textit{compatibility conditions} for the parameters \( \alpha, \beta \) and \( \gamma \) so that (52) admits a symmetry of fixed rank. The results are:

**Rank 3:** \( G = u_x \) is a symmetry of (52) without any conditions on the parameters.

**Rank 5:** \( G = uu_x + \frac{5}{3\gamma} u_{3x} \) is a symmetry of (52) provided that

\[ \alpha = \frac{3}{10} \gamma^2, \quad \text{and} \quad \beta = 2\gamma. \] (53)

The Lax equation satisfies (53). Since the KdV equation (5) is a member of Lax hierarchy, condition (53) comes as no surprise.

**Rank 7:** Equation (52) is of rank 7. The right hand side of (52) is the symmetry.

**Rank 9:** Three branches emerge:

(i) If condition (53) holds then

\[
G = u^3 u_x + \frac{5}{\gamma} u_x^3 + \frac{20}{\gamma} uu_x u_{2x} + \frac{5}{\gamma^2} u^2 u_{3x} + \frac{50}{\gamma^2} u_{2x} u_{3x} + \frac{30}{\gamma^2} u_{x} u_{4x}
\]

\[+ \frac{10}{\gamma^2} u u_{5x} + \frac{50}{\gamma^2} u_{5x}, \]

(ii) If

\[ \alpha = \frac{1}{5} \gamma^2, \quad \text{and} \quad \beta = \gamma \] (54) holds, one has the symmetry

\[
G = u^3 u_x + \frac{15}{4\gamma} u_x^3 + \frac{45}{2\gamma} uu_x u_{2x} + \frac{15}{2\gamma^2} u^2 u_{3x} + \frac{225}{4\gamma^2} u_{2x} u_{3x} + \frac{75}{2\gamma^2} u_{x} u_{4x}
\]

\[+ \frac{75}{4\gamma^2} u u_{5x} + \frac{375}{28\gamma^3} u_{7x}. \]

The SK equation satisfies the condition (54).

(iii) One has the symmetry

\[
G = u^3 u_x + \frac{75}{8\gamma} u_x^3 + \frac{135}{4\gamma} uu_x u_{2x} + \frac{15}{2\gamma^2} u^2 u_{3x} + \frac{225}{2\gamma^2} u_{2x} u_{3x} + \frac{525}{8\gamma^2} u_{x} u_{4x}
\]

\[+ \frac{75}{4\gamma^2} u u_{5x} + \frac{375}{28\gamma^3} u_{7x} \]

provided that

\[ \alpha = \frac{1}{5} \gamma^2, \quad \text{and} \quad \beta = \frac{5}{2} \gamma, \] (55)

which holds for the KK case.
Rank 11: One obtains the symmetry
\[ G = u^4 u_x + \frac{20}{\gamma} u u_x^3 + \frac{40}{\gamma^2} u^2 u_x u_{2x} + \frac{620}{3\gamma^3} u_x u_{2x}^2 + \frac{20}{3\gamma} u^3 u_{3x} + \frac{460}{3\gamma^2} u_x^2 u_{3x} \\
+ \frac{200}{\gamma^2} u u_x u_{3x} + \frac{120}{\gamma^3} u^2 u_{x} u_{4x} + \frac{400}{\gamma^3} u^3 u_{3x} u_{4x} + \frac{20}{\gamma^2} u^2 u_{5x} + \frac{800}{3\gamma^3} u_{2x} u_{5x} \\
+ \frac{800}{7\gamma^3} u u_{6x} + \frac{200}{7\gamma^3} u u_{7x} + \frac{1000}{63\gamma^4} u u_x \]

provided that the condition (53) for the Lax hierarchy is satisfied.

In summary, our algorithm allows one to ‘filter out’ all the integrable cases in the class (52). Alternatively, in [21] we investigated the conditions on the parameters $\alpha, \beta, \gamma$ such that (52) admits an infinite sequence (perhaps with gaps) of polynomial conservation laws. The conditions in [21] are exactly the same as the ones that emerge from the above symmetry analysis.

6 The Integrability Package

We now briefly describe the use of our Integrability Package, which has (among other things) the code for the computation of symmetries based on the algorithms in Sections 2 and 3. The Integrability Package is written in Mathematica [54] syntax. Users are assumed to have access to Mathematica 3.0. All the necessary files are available in MathSource [23] including on-line help, documentation, and built-in examples. The corresponding files should be put in the appropriate places on your platform. Detailed instructions are given in the documentation.

After launching Mathematica, type
\begin{verbatim}
In[1]:= <<Integrability`
\end{verbatim}
to read in the code. Doing so, you will get the statement:

\begin{verbatim}
Loading init.m for Integrability from AddOns.
\end{verbatim}

For the purpose of symmetry computations, the functions PDESymmetries and DDESymmetries are available.

Working with (35) as an example, the first two lines define the system ($r = q^*$), whereas the third line will produce the three symmetries listed in (36):
\begin{verbatim}
In[2]:= pde1 := D[q[x,t],t] - D[q[x,t],{x,2}] + 
     2*q[x,t]^2 r[x,t] == 0;

In[3]:= pde2 := D[r[x,t],t] + D[r[x,t],{x,2}] - 
     2*r[x,t]^2 q[x,t] == 0;
\end{verbatim}
In[4]:= PDESymmetries[{pde1,pde2},{q,r},{x,t},{4,6},  
                WeightRules->{Weight[q]->Weight[r]}]

Help about the key functions and their options can be obtained by typing

In[5]:= ??DDESymmetries

DDESymmetries[eqn, u, {n, t}, R, opts] finds the symmetry with rank
R of a differential-difference equation for the function u.
DDESymmetries[{eqn1, eqn2, ...}, {u1, u2, ...}, {n, t}, R, opts]
finds the symmetry of a system of differential-difference
equations, where R is the rank of the first equation in the
desired symmetry. DDESymmetries[{eqn1, eqn2, ...},
{u1, u2, ...}, {n, t}, {Rmax}, opts] finds the symmetries with
rank 0 through Rmax. DDESymmetries[{eqn1, eqn2, ...},
{u1, u2, ...}, {n, t}, {Rmin, Rmax}, opts] finds the symmetries
with rank Rmin through Rmax. n is understood as the discrete
space variable and t as the time variable.

Attributes[DDESymmetries] = {Protected, ReadProtected}

Options[DDESymmetries] =
   {WeightedParameters -> {}, WeightRules -> {}},
   MaxExplicitDependency -> 0, UndeterminedCoefficients -> C}

and

In[6]:= ??WeightedParameters

WeightedParameters is an option that determines the parameters with
weight. If WeightedParameters -> {p1, p2, ...},
then p1, p2, ... are considered as constant parameters with
weight. The default is WeightedParameters -> {}.

Attributes[WeightedParameters] = {Protected}

The option **WeightedParameters** is useful when working with systems that
are not uniform in rank. In such cases, the code tries to resolve the problem of
lack of uniformity, and prints appropriate messages. If the code can not auto-
matically resolve the problem it suggests the use of the **WeightedParameters**
option. Therefore, one should not use the option **WeightedParameters** unless
it is explicitly suggested. For further descriptions of these key functions and their
options we refer to the documentation in [23].
7 Other Software Packages

Higher-order symmetries can be computed with prolongation methods and numerous software packages are available that can aid in the tedious computations inherent to such methods. An extensive review of software for Lie symmetry computations, including generalized symmetries, can be found in [26, 27].

With prolongation methods one generates and subsequently reduces and solves a determining system of linear homogeneous partial differential equations for the coefficients of the unknown higher-order symmetry generator. In many cases, due to the length and complexity of that system, the general solution is out of reach and one quite often resorts to making a polynomial ansatz for the symmetry.

Although restricted to polynomial higher-order symmetries, we believe that the method presented in this paper is much more straightforward. Furthermore, it avoids the application of prolongation methods or Lie algebraic techniques.

To avoid retreading the surveys [26, 27], we restrict our discussion to algorithms and symbolic packages that allow one to compute generalized symmetries of PDEs, as they were defined in Section 2.1. We are not aware of software packages for DDEs to calculate directly the type of symmetries defined in Section 3.1.

Yamilov and co-workers [11, 38, 49, 50, 55] carried out an integrability study based on formal symmetries for DDEs. They explicitly mention that their symmetry algorithms are suitable for implementation in computer algebra systems, but as far as we know they have not yet implemented them. The computation of classical and non-classical Lie-point symmetries based on prolongation techniques is addressed in e.g. [39].

Based on the alternative strategy discussed in Remark (ii) in Section 2, Ito’s programs in REDUCE [32, 33] compute polynomial higher-order symmetries for systems of evolution equations that are uniform in rank (no weighted parameters can be introduced). Ito’s programs can not be used to computed symmetries that explicitly depend on $x$ and $t$.

In [17], Fuchssteiner et al. present an algorithm to compute higher-order symmetries of evolution equations. Their algorithm is based on Lie algebraic techniques and uses commutator algebra on the Lie algebra of vector fields. Their approach is different from the usual prolongation method in that no determining equations are solved. Instead, all necessary generators of the finitely generated Virasoro algebra are computed from one given element by direct Lie algebra methods. Their code is available in MuPAD.

The mastersymmetry approach of Fuchssteiner and collaborators has been applied to systems of DDEs in [41, 56]. Computing mastersymmetries is algorithmic yet cumbersome. With the help of Maple and MuPAD and the recursive mastersymmetry procedure, they were able to determine the explicit form of master-
symmetries for a variety of DDE systems [41, 56]. Based on the knowledge of one non-trivial mastersymmetry, they obtained conserved densities, symmetries, recursion operators, and multi-Hamiltonian formulations for many integrable DDEs.

The REDUCE program FS for “formal symmetries” was written by Gerdt and Zharkov [19]. The code FS can be applied to polynomial nonlinear PDEs of evolution type, which are linear with respect to the highest-order spatial derivatives and with non-degenerated, diagonal coefficient matrix for the highest derivatives. The algorithm in FS requires that the evolution equation are of order two or higher in the spatial variable. However, this approach does not require that the evolution equations are uniform in rank. With FS one cannot compute symmetries that depend explicitly on $x$ and $t$.

The PC package DELiA, written in Turbo PASCAL by Bocharov [10] and co-workers, is a commercial computer algebra system for investigating differential equations using Lie’s approach. The program deals with higher-order symmetries, conservation laws, integrability and equivalence problems. It has a special routine for systems of evolution equations. The program requires the presence of second or higher-order spatial derivative terms in all equations.

Finally, Sanders and Wang [46] have Maple and FORM software that aids in the computation of recursion operators.

8 Conclusions

We have implemented direct algorithms (in Mathematica) that allow the user to compute polynomial generalized (higher-order) symmetries of polynomial systems of evolution and lattice equations.

These algorithms are based on the dilation invariance of the given equations. Only minor modifications of our strategy lead to direct algorithms for conserved densities for systems of nonlinear PDEs [21] and nonlinear DDEs [22, 24].

For systems that arise from a variational principle, conservation laws follow from higher-order symmetries (Noether’s theorem) and vice versa. Currently, in our algorithms we are not exploiting such connections.

In the future we will investigate generalizations of our methods to PDEs and DDEs in multiple space dimensions. We will also study the use of Lie-point symmetries other than dilation symmetries. Moreover, most recursion operators, which connect generalized symmetries, are also dilation invariant. We will continue working on the extension of our algorithms to the symbolic computation of recursion operators [28]. A comparison of our algorithm with the formal symmetry and mastersymmetry approaches described in Section 6 is also part of future work.
Acknowledgements

We acknowledge helpful discussions with Profs. Clara Nucci, Peter Olver, Jan Sanders, and Pavel Winternitz.

References

[19] V. P. Gerdt and A. Y. Zharilov, Computer generation of necessary integrability conditions for polynomial-nonlinear evolution systems, in Proc. ISSAC ’90, S. Watanabe and M.
Ü. Göktaş and W. Hereman / Symbolic Computation of Symmetries


[41] W. Oevel, H. Zhang, G. Tu and B. Fuchsteiner, Mastersymmetries and multi-Hamiltonian