Centre de Recherches Mathématiques
CRM Proceedings and Lecture Notes

Symbolic Computation of Conserved Densities, Generalized Symmetries, and Recursion Operators for Nonlinear Differential-Difference Equations

Willy Hereman, Jan A. Sanders, Jack Sayers, and Jing Ping Wang

This paper is dedicated to Ryan Sayers (1982-2003)

Abstract. Algorithms for the symbolic computation of conserved densities, fluxes, generalized symmetries, and recursion operators for systems of nonlinear differential-difference equations are presented. In the algorithms we use discrete versions of the Fréchet and variational derivatives, as well as discrete Euler and homotopy operators.

The algorithms are illustrated for prototypical nonlinear lattices, including the Kac-van Moerbeke (Volterra) and Toda lattices. Results are shown for the modified Volterra and Ablowitz-Ladik lattices.

1. Introduction

The study of complete integrability of nonlinear differential-difference equations (DDEs) largely parallels that of PDEs [**AC91, ASY00, W98**]. Indeed, as in the continuous case, the existence of sufficiently many conserved densities and generalized symmetries is a predictor for complete integrability. Based on the first few densities and symmetries a recursion operator (which maps symmetries to symmetries) can be constructed. The existence of a recursion operator, which allows one to generate an infinite sequence of symmetries, confirms complete integrability.

There is a vast body of work on the complete integrability of DDEs. Consult e.g. [**ASY00, HH03**] for various approaches and references. In this article we describe algorithms to symbolically compute polynomial conservation laws, fluxes, generalized symmetries, and recursion operators for DDEs. The design of these algorithms heavily relies on related work for PDEs [**GH97a, HG99, HGCM98**] and work by Oevel *et al* [**OZF93**].

There exists a close analogy between the continuous and discrete (in space) cases. As shown in [**HM02, MH02**], this analogy can be completely formalized

1991 *Mathematics Subject Classification.* Primary 37J35, 37K10; Secondary 35Q58, 37K05.

Key words and phrases. Homotopy operators, conserved densities, fluxes, recursion operators, integrability, differential-difference, DDEs, semi-discrete lattices.

This material is based upon work supported by the National Science Foundation under Grant No. CCR-9901929.

and both theories can be formulated in terms of complexes. The same applies for the formulation in terms of Lie algebra complexes in [**W98**]. This allows one to translate by analogy the existing algorithms immediately (although complications arise when there is explicit dependence on the space variable in the discrete case). One of the more useful tools following from the abstract theory is the homotopy operator, which is based on scaling vectorfields, and goes back to Poincaré in the continuous case. This operator allows one to directly integrate differential forms and can be straightforwardly implemented in computer algebra packages, since it reduces to integration over one scaling parameter. The discrete analogue as described in [**HM02, MH02**] does the corresponding job in the discrete case and we use it to compute fluxes. In this paper we do not explicitly use the abstract framework, yet, it has been a motivating force for the development of our algorithms.

The algorithms in this paper can be implemented in any computer algebra system. Our Mathematica package *InvariantsSymmetries.m* [**GH97b**] computes densities and symmetries, and therefore aids in automated testing of complete integrability of semi-discrete lattices. Mathematica code to automatically compute recursion operators is still under development.

The paper is organized as follows. In Section 2, we give key definitions and introduce the Kac-van Moerbeke (KvM) [**KvM75**], Toda [**T81**] and Ablowitz-Ladik (AL) lattices [**AL75**], which will be used as prototypical examples throughout the paper. The discrete higher Euler operators (variational derivatives) and the discrete homotopy operator are introduced in Section 3. These operators are applied in the construction of densities and fluxes in Section 5. The algorithm for higher-order symmetries is outlined in Section 6. The algorithms for scalar and matrix recursion operators are given in Section 7 and Section 8. The paper concludes with two more examples in Section 9.

2. Key Definitions and Prototypical Examples

Definition 2.1. A nonlinear (autonomous) DDE is an equation of the form

$$\dot{\mathbf{u}}_n = \mathbf{F}(..., \mathbf{u}_{n-1}, \mathbf{u}_n, \mathbf{u}_{n+1}, ...), \tag{2.1}$$

where $\mathbf{u}_n$ and $\mathbf{F}$ are vector-valued functions with m components. The integer n corresponds to discretization in space; the dot denotes one derivative with respect to the continuous time variable (t).

For simplicity, we will denote the components of $\mathbf{u}_n$ by $(u_n, v_n, w_n, \cdots)$. For brevity, we write $\mathbf{F}(\mathbf{u}_n)$, although $\mathbf{F}$ typically depends on $\mathbf{u}_n$ and a finite number of its forward and backward shifts. We assume that $\mathbf{F}$ is polynomial with constant coefficients. If present, parameters are denoted by lower-case Greek letters. No restrictions are imposed on the level of the forward or backward shifts or the degree of nonlinearity in $\mathbf{F}$.

Example 2.2. The AL lattice [**AL75**],

$$\begin{aligned} \dot{u}_n &= (u_{n+1} - 2u_n + u_{n-1}) + u_n v_n (u_{n+1} + u_{n-1}), \\ \dot{v}_n &= -(v_{n+1} - 2v_n + v_{n-1}) - u_n v_n (v_{n+1} + v_{n-1}), \end{aligned} \tag{2.2}$$

is a completely integrable discretization of the nonlinear Schrödinger equation.

Definition 2.3. A DDE is said to be dilation invariant if it is invariant under a scaling (dilation) symmetry.

EXAMPLE 2.4. The KvM lattice [**KvM75**],

$$\dot{u}_n = u_n(u_{n+1} - u_{n-1}), \tag{2.3}$$

is invariant under $(t, u_n) \to (\lambda^{-1}t, \lambda u_n)$, where λ is an arbitrary scaling parameter.

EXAMPLE 2.5. The Toda lattice [**T81**] in polynomial form [**GH98**],

$$\dot{u}_n = v_{n-1} - v_n, \quad \dot{v}_n = v_n(u_n - u_{n+1}), \tag{2.4}$$

is invariant under the scaling symmetry

$$(t, u_n, v_n) \to (\lambda^{-1}t, \lambda u_n, \lambda^2 v_n). \tag{2.5}$$

Thus, u_n and v_n correspond to one, respectively two derivatives with respect to t,

$$u_n \sim \frac{\mathrm{d}}{\mathrm{dt}}, \quad v_n \sim \frac{\mathrm{d}^2}{\mathrm{dt}^2}. \tag{2.6}$$

DEFINITION 2.6. We define the weight, w, of a variable as the number of t-derivatives the variable corresponds to.

Since t is replaced by $\frac{t}{\lambda}$, we set $w(\frac{\mathrm{d}}{\mathrm{dt}}) = 1$. In view of (2.6), we have $w(u_n) = 1$, and $w(v_n) = 2$ for the Toda lattice. Weights of dependent variables are nonnegative, rational, and independent of n.

DEFINITION 2.7. The rank of a monomial is defined as the total weight of the monomial. An expression is uniform in rank if all its terms have the same rank.

EXAMPLE 2.8. In the first equation of (2.4), all the monomials have rank 2; in the second equation all the monomials have rank 3. Conversely, requiring uniformity in rank for each equation in (2.4) allows one to compute the weights of the dependent variables with simple linear algebra. Indeed,

$$w(u_n) + 1 = w(v_n), \quad w(v_n) + 1 = w(u_n) + w(v_n), \tag{2.7}$$

yields

$$w(u_n) = 1, \quad w(v_n) = 2, \tag{2.8}$$

which is consistent with (2.5).

Dilation symmetries, which are special Lie-point symmetries, are common to many lattice equations. Lattices that do not admit a dilation symmetry can be made scaling invariant by extending the set of dependent variables using auxiliary parameters with scaling.

EXAMPLE 2.9. The AL lattice is not dilation invariant. Introducing an auxiliary parameter α, hence replacing (2.2) by

$$\begin{aligned} \dot{u}_n &= \alpha(u_{n+1} - 2u_n + u_{n-1}) + u_n v_n(u_{n+1} + u_{n-1}), \\ \dot{v}_n &= -\alpha(v_{n+1} - 2v_n + v_{n-1}) - u_n v_n(v_{n+1} + v_{n-1}), \end{aligned} \tag{2.9}$$

and requiring uniformity in rank, gives

$$\begin{aligned} w(u_n) + 1 &= w(\alpha) + w(u_n) = 2w(u_n) + w(v_n), \\ w(v_n) + 1 &= w(\alpha) + w(v_n) = 2w(v_n) + w(u_n). \end{aligned} \tag{2.10}$$

Obviously,

$$w(u_n) + w(v_n) = w(\alpha) = 1. \tag{2.11}$$

Several scales are possible. The choice $w(u_n) = w(v_n) = \frac{1}{2}, w(\alpha) = 1$, corresponds to the scaling symmetry

$$(t, u_n, v_n, \alpha) \to (\lambda^{-1} t, \lambda^{\frac{1}{2}} u_n, \lambda^{\frac{1}{2}} v_n, \lambda\alpha). \tag{2.12}$$

Definition 2.10. A scalar function $\rho_n(\mathbf{u}_n)$ is a conserved density of (2.1) if there exists a scalar function $J_n(\mathbf{u}_n)$ called the associated flux, such that

$$\mathrm{D}_t \rho_n + \Delta\, J_n = 0 \tag{2.13}$$

is satisfied on the solutions of (2.1) [**O93**].

In (2.13), we used the (forward) difference operator, $\Delta = \mathrm{D} - \mathrm{I}$, defined by

$$\Delta\, J_n = (\mathrm{D} - \mathrm{I})\, J_n = J_{n+1} - J_n, \tag{2.14}$$

where D denotes the up-shift (forward or right-shift) operator, $\mathrm{D} J_n = J_{n+1}$, and I is the identity operator. Operator Δ takes the role of a spatial derivative on the shifted variables as many DDEs arise from discretization of a PDE in $(1+1)$ variables. Most, but not all, densities are polynomial in $\mathbf{u}_n$.

Example 2.11. The first three conservation laws for (2.3) are

$$\mathrm{D}_t(\ln(u_n)) + (u_{n+1} + u_n) - (u_n + u_{n-1}) = 0, \tag{2.15}$$

$$\mathrm{D}_t(u_n) + (u_{n+1} u_n) - (u_n u_{n-1}) = 0, \tag{2.16}$$

$$\mathrm{D}_t(\frac{1}{2} u_n^2 + u_n u_{n+1}) + u_n u_{n+1}(u_{n+1} + u_{n+2}) - u_{n-1} u_n (u_n + u_{n+1}) = 0. \tag{2.17}$$

Example 2.12. For the Toda lattice (2.4) the first four density-flux pairs are

$$\rho_n^{(0)} = \ln(v_n), \qquad J_n^{(0)} = u_n, \tag{2.18}$$

$$\rho_n^{(1)} = u_n, \qquad J_n^{(1)} = v_{n-1}, \tag{2.19}$$

$$\rho_n^{(2)} = \frac{1}{2} u_n^2 + v_n, \qquad J_n^{(2)} = u_n v_{n-1}, \tag{2.20}$$

$$\rho_n^{(3)} = \frac{1}{3} {u_n}^3 + u_n(v_{n-1} + v_n), \qquad J_n^{(3)} = u_{n-1} u_n v_{n-1} + {v_{n-1}}^2. \tag{2.21}$$

The above densities are uniform of ranks 0 through 3. The corresponding fluxes are also uniform in rank with ranks 1 through 4. In general, if in (2.13) rank $\rho_n = R$ then rank $J_n = R + 1$, since $w(\mathrm{D}_t) = 1$.

This comes as no surprise since the conservation law (2.13) holds on solutions of (2.1), hence it 'inherits' the dilation symmetry of (2.1).

In Section 5 we will give an algorithm to compute polynomial conserved densities and fluxes and use (2.4) to illustrate the steps. Non-polynomial densities (which are easy to find by hand) can be computed with the method given in [**HH03**].

Definition 2.13. Compositions of D and D^{-1} define an equivalence relation ($\equiv$) on monomials in the components of $\mathbf{u}_n$. All shifted monomials are equivalent.

Example 2.14. For example, $u_{n-1} v_{n+1} \equiv u_{n+2} v_{n+4} \equiv u_{n-3} v_{n-1}$.

Definition 2.15. The main representative of an equivalence class is the monomial of the class with n as lowest label on any component of u (or v if u is missing).

Example 2.16. For example, $u_n u_{n+2}$ is the main representative of the class $\{\cdots, u_{n-2} u_n, u_{n-1} u_{n+1}, u_n u_{n+2}, u_{n+1} u_{n+3}, \cdots\}$. Use lexicographical ordering to resolve conflicts. For example, $u_n v_{n+2}$ (not $u_{n-2} v_n$) is the main representative of the class $\{\cdots, u_{n-3} v_{n-1}, \cdots, u_{n+2} v_{n+4}, \cdots\}$.

DEFINITION 2.17. A vector function $\mathbf{G}(\mathbf{u}_n)$ is called a generalized symmetry of (2.1) if the infinitesimal transformation $\mathbf{u}_n \to \mathbf{u}_n + \epsilon \mathbf{G}$ leaves (2.1) invariant up to order ϵ. Consequently, $\mathbf{G}$ must satisfy [**O93**]

$$\mathrm{D}_t \mathbf{G} = \mathbf{F}'(\mathbf{u}_n)[\mathbf{G}], \tag{2.22}$$

on solutions of (2.1). $\mathbf{F}'(\mathbf{u}_n)[\mathbf{G}]$ is the Fréchet derivative of $\mathbf{F}$ in the direction of $\mathbf{G}$.

For the scalar case ($N = 1$), the Fréchet derivative in the direction of G is

$$F'(u_n)[G] = \frac{\partial}{\partial \epsilon} F(u_n + \epsilon G)|_{\epsilon=0} = \sum_k \frac{\partial F}{\partial u_{n+k}} \mathrm{D}^k G, \tag{2.23}$$

which defines the Fréchet derivative operator

$$F'(u_n) = \sum_k \frac{\partial F}{\partial u_{n+k}} \mathrm{D}^k. \tag{2.24}$$

In the vector case with two components, u_n and v_n, the Fréchet derivative operator is

$$\mathbf{F}'(\mathbf{u}_n) = \begin{pmatrix} \sum_k \frac{\partial F_1}{\partial u_{n+k}} \mathrm{D}^k & \sum_k \frac{\partial F_1}{\partial v_{n+k}} \mathrm{D}^k \\ \sum_k \frac{\partial F_2}{\partial u_{n+k}} \mathrm{D}^k & \sum_k \frac{\partial F_2}{\partial v_{n+k}} \mathrm{D}^k \end{pmatrix}. \tag{2.25}$$

Applied to $\mathbf{G} = (G_1 \quad G_2)^{\mathrm{T}}$, where T is transpose, one obtains

$$F_i{}'(\mathbf{u}_n)[\mathbf{G}] = \sum_k \frac{\partial F_i}{\partial u_{n+k}} \mathrm{D}^k G_1 + \sum_k \frac{\partial F_i}{\partial v_{n+k}} \mathrm{D}^k G_2, \quad i = 1, 2. \tag{2.26}$$

In (2.23) and (2.26) summation is over all positive and negative shifts (including $k = 0$). For $k > 0$, $\mathrm{D}^k = \mathrm{D} \circ \mathrm{D} \circ \cdots \circ \mathrm{D}$ (k times). Similarly, for $k < 0$ the down-shift operator D^{-1} is applied repeatedly. The generalization of (2.25) to a system with N components should be obvious.

EXAMPLE 2.18. The first two symmetries of (2.3) are

$$G^{(1)} = u_n(u_{n+1} - u_{n-1}), \tag{2.27}$$

$$G^{(2)} = u_n u_{n+1}(u_n + u_{n+1} + u_{n+2}) - u_{n-1} u_n (u_{n-2} + u_{n-1} + u_n). \tag{2.28}$$

These symmetries are uniform in rank (rank 2 and 3, respectively). The symmetries of ranks 0 and 1 are both zero.

EXAMPLE 2.19. The first two non-trivial symmetries of (2.4) are

$$\mathbf{G}^{(1)} = \begin{pmatrix} v_{n-1} - v_n \\ v_n(u_n - u_{n+1}) \end{pmatrix}, \tag{2.29}$$

and

$$\mathbf{G}^{(2)} = \begin{pmatrix} v_n(u_n + u_{n+1}) - v_{n-1}(u_{n-1} + u_n) \\ v_n(u_{n+1}^2 - u_n^2 + v_{n+1} - v_{n-1}) \end{pmatrix}. \tag{2.30}$$

The above symmetries are uniform in rank. For example, $\operatorname{rank} G_1^{(2)} = 3$ and $\operatorname{rank} G_2^{(2)} = 4$. Symmetries of lower ranks are trivial.

An algorithm to compute generalized symmetries will be outlined in Section 6 and applied to (2.4).

DEFINITION 2.20. A recursion operator $\mathcal{R}$ connects symmetries

$$\mathbf{G}^{(j+s)} = \mathcal{R}\mathbf{G}^{(j)}, \tag{2.31}$$

where $j = 1, 2, ...,$ and s is the seed. In most cases the symmetries are consecutively linked ($s = 1$). For N-component systems, $\mathcal{R}$ is an $N \times N$ matrix operator.

The defining equation for $\mathcal{R}$ [**O93, W98**] is

$$\mathrm{D}_t\mathcal{R} + [\mathcal{R}, \mathbf{F}'(\mathbf{u}_n)] = \frac{\partial \mathcal{R}}{\partial t} + \mathcal{R}'[\mathbf{F}] + \mathcal{R} \circ \mathbf{F}'(\mathbf{u}_n) - \mathbf{F}'(\mathbf{u}_n) \circ \mathcal{R} = 0, \tag{2.32}$$

where [,] denotes the commutator and $\circ$ the composition of operators. The operator $\mathbf{F}'(\mathbf{u}_n)$ was defined in (2.25). $\mathcal{R}'[\mathbf{F}]$ is the Fréchet derivative of $\mathcal{R}$ in the direction of $\mathbf{F}$. For the scalar case, the operator $\mathcal{R}$ is often of the form

$$\mathcal{R} = U(u_n)\, \mathcal{O}((\mathrm{D} - \mathrm{I})^{-1}, \mathrm{D}^{-1}, \mathrm{I}, \mathrm{D})\, V(u_n), \tag{2.33}$$

and then

$$\mathcal{R}'[F] = \sum_k (\mathrm{D}^k F)\, \frac{\partial U}{\partial u_{n+k}}\, \mathcal{O}\, V + \sum_k U\mathcal{O}\, (\mathrm{D}^k F)\, \frac{\partial V}{\partial u_{n+k}}. \tag{2.34}$$

For the vector case, the elements of the $N \times N$ operator matrix $\mathcal{R}$ are often of the form $\mathcal{R}_{ij} = U_{ij}(\mathbf{u}_n)\, \mathcal{O}_{ij}((\mathrm{D} - \mathrm{I})^{-1}, \mathrm{D}^{-1}, \mathrm{I}, \mathrm{D})\, V_{ij}(\mathbf{u}_n)$. For the two-component case

$$\begin{aligned} \mathcal{R}'[\mathbf{F}]_{ij} \;&=\; \sum_k (\mathrm{D}^k F_1)\, \frac{\partial U_{ij}}{\partial u_{n+k}}\, \mathcal{O}_{ij}\, V_{ij} + \sum_k (\mathrm{D}^k F_2)\, \frac{\partial U_{ij}}{\partial v_{n+k}}\, \mathcal{O}_{ij}\, V_{ij} \\ &\quad + \sum_k U_{ij}\mathcal{O}_{ij}\, (\mathrm{D}^k F_1)\, \frac{\partial V_{ij}}{\partial u_{n+k}} + \sum_k U_{ij}\mathcal{O}_{ij}\, (\mathrm{D}^k F_2)\, \frac{\partial V_{ij}}{\partial v_{n+k}}. \end{aligned} \tag{2.35}$$

EXAMPLE 2.21. The KvM lattice (2.3) has recursion operator

$$\begin{aligned} \mathcal{R} \;&=\; u_n(\mathrm{I} + \mathrm{D})(u_n\mathrm{D} - \mathrm{D}^{-1}u_n)(\mathrm{D} - \mathrm{I})^{-1}\frac{1}{u_n}\mathrm{I} \\ &=\; u_n\mathrm{D}^{-1} + (u_n + u_{n+1})\mathrm{I} + u_n\mathrm{D} + u_n(u_{n+1} - u_{n-1})(\mathrm{D} - \mathrm{I})^{-1}\frac{1}{u_n}\mathrm{I}. \end{aligned} \tag{2.36}$$

EXAMPLE 2.22. The Toda lattice (2.4) has recursion operator

$$\mathcal{R} = \begin{pmatrix} -u_n\mathrm{I} & -\mathrm{D}^{-1} - \mathrm{I} + (v_{n-1} - v_n)(\mathrm{D} - \mathrm{I})^{-1}\frac{1}{v_n}\mathrm{I} \\ -v_n\mathrm{I} - v_n\mathrm{D} & -u_{n+1}\mathrm{I} + v_n(u_n - u_{n+1})(\mathrm{D} - \mathrm{I})^{-1}\frac{1}{v_n}\mathrm{I} \end{pmatrix}. \tag{2.37}$$

In Section 7 we will give an algorithm for the computation of scalar recursion operators like (2.36). In Section 8 we cover the matrix case and show how (2.37) is computed. The algorithms complement those for recursion operators of PDEs presented in [**HG99**] and elsewhere in these proceedings [**BHS04**]. We now introduce two powerful tools which will be used in the computation of densities and fluxes.

3. The Discrete Variational Derivative (Euler Operator)

DEFINITION 3.1. A function $E_n(\mathbf{u}_n)$ is a total difference if there exists another function $J_n(\mathbf{u}_n)$, such that $E_n = \Delta\, J_n = (\mathrm{D} - \mathrm{I})\, J_n$.

THEOREM 3.2. *A necessary and sufficient condition for a function E_n with positive shifts up to level p_0, to be a total difference is that*

$$\mathcal{L}^{(0)}_{\mathbf{u}_n}(E_n) = 0, \tag{3.1}$$

where $\mathcal{L}^{(0)}_{\mathbf{u}_n}$ *is the discrete variational derivative (Euler operator)* [**ASY00**] *defined by*

$$\mathcal{L}^{(0)}_{\mathbf{u}_n} = \frac{\partial}{\partial \mathbf{u}_n}\left(\sum_{k=0}^{p_0} \mathrm{D}^{-k}\right) = \frac{\partial}{\partial \mathbf{u}_n}(\mathrm{I} + \mathrm{D}^{-1} + \mathrm{D}^{-2} + \cdots + \mathrm{D}^{-p_0}). \tag{3.2}$$

A proof is given in e.g. [**HH03**].

Remark 3.3. To verify that an expression $E(u_{n-q}, \cdots, u_n, \cdots, u_{n+p})$ involving negative shifts is a total difference, one must first remove the negative shifts by replacing E_n by $\tilde{E}_n = \mathrm{D}^q E_n$. Applied to $\tilde{E}_n$, (3.2) terminates at $p_0 = p + q$.

We now introduce a tool to invert the total difference operator $\Delta = \mathrm{D} - \mathrm{I}$.

4. The Discrete Homotopy Operator

Given is an expression E_n (free of negative shifts). We assume that one has verified that $E_n \in \operatorname{Ker} \mathcal{L}^{(0)}_{\mathbf{u}_n}$, i.e. $\mathcal{L}^{(0)}_{\mathbf{u}_n}(E_n) = 0$. Consequently, $E_n \in \operatorname{Im} \Delta$. So, $\exists J_n$ such that $E_n = \Delta J_n$. To compute $J_n = \Delta^{-1}(E_n)$ one must invert the operator $\Delta = \mathrm{D} - \mathrm{I}$. Working with the formal inverse,

$$\Delta^{-1} = \mathrm{D}^{-1} + \mathrm{D}^{-2} + \mathrm{D}^{-3} + \ldots, \tag{4.1}$$

is impractical, perhaps impossible. We therefore present the (discrete) homotopy operator which circumvents the above infinite series. In analogy to the continuous case [**O93**], we first introduce the discrete higher Euler operators.

Definition 4.1. The discrete higher Euler operators are defined by

$$\mathcal{L}^{(i)}_{\mathbf{u}_n} = \frac{\partial}{\partial \mathbf{u}_n}\left(\sum_{k=i}^{p_i} \binom{k}{i} \mathrm{D}^{-k}\right) \tag{4.2}$$

The higher Euler operators all terminate at some maximal shifts p_i.

Example 4.2. For scalar component u_n, the first higher Euler operators are:

$$\mathcal{L}^{(0)}_{u_n} = \frac{\partial}{\partial u_n}(\mathrm{I} + \mathrm{D}^{-1} + \mathrm{D}^{-2} + \mathrm{D}^{-3} + \cdots + \mathrm{D}^{-p_0}), \tag{4.3}$$

$$\mathcal{L}^{(1)}_{u_n} = \frac{\partial}{\partial u_n}(\mathrm{D}^{-1} + 2\mathrm{D}^{-2} + 3\mathrm{D}^{-3} + 4\mathrm{D}^{-4} + \cdots + p_1\,\mathrm{D}^{-p_1}), \tag{4.4}$$

$$\mathcal{L}^{(2)}_{u_n} = \frac{\partial}{\partial u_n}(\mathrm{D}^{-2} + 3\mathrm{D}^{-3} + 6\mathrm{D}^{-4} + 10\mathrm{D}^{-5} + \cdots + \frac{1}{2}p_2(p_2 - 1)\,\mathrm{D}^{-p_2}). \tag{4.5}$$

Note that (4.3) coincides with (3.2). Similar formulae hold for $\mathcal{L}^{(i)}_{v_n}$.

Next, we introduce the homotopy operators. For notational simplicity we show the formulae for the two component case $\mathbf{u}_n = (u_n, v_n)$.

Definition 4.3. The total homotopy operator is defined as

$$\mathcal{H} = \int_0^1 (j_{1,n}(\mathbf{u}_n)[\lambda \mathbf{u}_n] + j_{2,n}(\mathbf{u}_n)[\lambda \mathbf{u}_n])\frac{d\lambda}{\lambda}, \tag{4.6}$$

where the homotopy operators are given by

$$j_{1,n}(\mathbf{u}_n) = \sum_{i=0}^{p_1 - 1} (\mathrm{D} - \mathrm{I})^i (u_n \mathcal{L}^{(i+1)}_{u_n}), \quad j_{2,n}(\mathbf{u}_n) = \sum_{i=0}^{p_2 - 1} (\mathrm{D} - \mathrm{I})^i (v_n \mathcal{L}^{(i+1)}_{u_n}). \tag{4.7}$$

THEOREM 4.4. *$J_n = \Delta^{-1}(E_n)$ can be computed as $J_n = \mathcal{H}(E_n)$.*

A similar theorem and proof for continuous homotopy operators is given in [**O93**]. By constructing a similar variational bicomplex the theorem still holds in the discrete case. See [**HM02, MH02**] and elsewhere in these proceedings [**M04**].

REMARK 4.5. Since the theory for the recursion operator can be formulated in the language of Lie algebra complexes, the results in [**SW01a**] are immediately applicable. This gives one explicit conditions from which the existence of infinite hierarchies of local symmetries and cosymmetries can be concluded. Also, the problems with the definition of a recursion operator as treated in [**SW01b**] also play a role in the discrete case.

REMARK 4.6. Note that p_1 and p_2 in (4.7) are the highest shifts for u_n and v_n in E_n. Furthermore, $j_{r,n}(\mathbf{u}_n)[\lambda\mathbf{u}_n]$ means that in $j_{r,n}(\mathbf{u}_n)$ one replaces $\mathbf{u}_n \to \lambda\mathbf{u}_n$, and, of course, $\mathbf{u}_{n+1} \to \lambda\mathbf{u}_{n+1}$, $\mathbf{u}_{n+2} \to \lambda\mathbf{u}_{n+2}$, etc.

REMARK 4.7. In practice, one does not compute the definite integral (4.6) since the evaluation at boundary $\lambda = 0$ may cause problems. Instead, one computes the indefinite integral and evaluates the result at $\lambda = 1$.

5. Algorithm to Compute Densities and Fluxes

As an example, we compute the density-flux pair (2.21) for (2.4). Assuming that the weights (2.8) are computed and the rank of the density is selected ($R = 3$ here), the algorithm has three steps.

Step 1: Construct the form of the density

List all monomials in u_n and v_n of rank 3 or less: $\mathcal{G} = \{{u_n}^3, {u_n}^2, u_n v_n, u_n, v_n\}$.

Next, for each monomial in $\mathcal{G}$, introduce the correct number of t-derivatives so that each term has weight 3. Thus, using (2.4),

$$\begin{aligned}
&\frac{\mathrm{d}^0}{\mathrm{d}t^0}({u_n}^3) = {u_n}^3, \qquad \frac{\mathrm{d}^0}{\mathrm{d}t^0}(u_n v_n) = u_n v_n,\\
&\frac{\mathrm{d}}{\mathrm{d}t}({u_n}^2) = 2u_n\dot{u}_n = 2u_n v_{n-1} - 2u_n v_n, \qquad \frac{\mathrm{d}}{\mathrm{d}t}(v_n) = \dot{v}_n = u_n v_n - u_{n+1} v_n,\\
&\frac{\mathrm{d}^2}{\mathrm{d}t^2}(u_n) = \frac{\mathrm{d}}{\mathrm{d}t}(\dot{u}_n) = \frac{\mathrm{d}}{\mathrm{d}t}(v_{n-1} - v_n) = u_{n-1}v_{n-1} - u_n v_{n-1} - u_n v_n + u_{n+1} v_n.
\end{aligned}$$

Gather all terms in $\mathcal{H} = \{{u_n}^3, u_n v_{n-1}, u_n v_n, u_{n-1} v_{n-1}, u_{n+1} v_n\}$. Identify members belonging to the same equivalence classes and replace them by their main representatives. For example, $u_n v_{n-1} \equiv u_{n+1} v_n$, so both are replaced by $u_n v_{n-1}$. So, $\mathcal{H}$ is replaced by $\mathcal{I} = \{{u_n}^3, u_n v_{n-1}, u_n v_n\}$, which has the building blocks of the density. Linear combination of the monomials in $\mathcal{I}$ with constant coefficients c_i gives

$$\rho_n = c_1\, {u_n}^3 + c_2\, u_n v_{n-1} + c_3\, u_n v_n. \tag{5.1}$$

Step 2: Determine the coefficients

Require that (2.13) holds. Compute $\mathrm{D}_t\rho_n$. Use (2.4) to remove $\dot{u}_n$ and $\dot{v}_n$ and their shifts. Thus,

$$\begin{aligned}
E_n \;=\; & \mathrm{D}_t\rho_n = (3c_1 - c_2)u_n^2 v_{n-1} + (c_3 - 3c_1)u_n^2 v_n + (c_3 - c_2)v_{n-1}v_n\\
& + c_2 u_{n-1}u_n v_{n-1} + c_2 v_{n-1}^2 - c_3 u_n u_{n+1} v_n - c_3 v_n^2.
\end{aligned} \tag{5.2}$$

Compute $\tilde{E}_n = \mathrm{D}E_n$. First, apply (3.2) for component u_n to $\tilde{E}_n$:

$$\begin{aligned} \mathcal{L}^{(0)}_{u_n}(\tilde{E}_n) &= \frac{\partial}{\partial u_n}(\mathrm{I}+\mathrm{D}^{-1}+\mathrm{D}^{-2})(\tilde{E}_n) \\ &= 2(3c_1-c_2)u_nv_{n-1}+2(c_3-3c_1)u_nv_n \\ &\quad +(c_2-c_3)u_{n-1}v_{n-1}+(c_2-c_3)u_{n+1}v_n. \end{aligned} \tag{5.3}$$

Second, apply (3.2) for component v_n to $\tilde{E}_n$:

$$\begin{aligned} \mathcal{L}^{(0)}_{v_n}(\tilde{E}_n) &= \frac{\partial}{\partial v_n}(\mathrm{I}+\mathrm{D}^{-1})(\tilde{E}_n) \\ &= (3c_1-c_2)u_{n+1}^2+(c_3-c_2)v_{n+1}+(c_2-c_3)u_nu_{n+1} \\ &\quad +2(c_2-c_3)v_n+(c_3-3c_1)u_n^2+(c_3-c_2)v_{n-1}. \end{aligned} \tag{5.4}$$

Both (5.3) and (5.4) must vanish identically. Solve the linear system

$$\mathcal{S} = \{3c_1-c_2=0, c_3-3c_1=0, c_2-c_3=0\}. \tag{5.5}$$

The solution is $3c_1 = c_2 = c_3$. Substituting $c_1=\frac{1}{3}, c_2=c_3=1$, into (5.1)

$$\rho_n = \frac{1}{3}{u_n}^3 + u_n(v_{n-1}+v_n). \tag{5.6}$$

Step 3: **Compute the flux**

In view of (2.13), we will compute $J_n = \Delta^{-1}(-E_n)$ with the homotopy operator introduced in Section 4. Alternative methods are described in [**GH98, HH03**].

Insert $c_1=\frac{1}{3}, c_2=c_3=1$ into (5.2) and compute

$$\tilde{E}_n = \mathrm{D}E_n = u_nu_{n+1}v_n + v_n^2 - u_{n+1}u_{n+2}v_{n+1} - v_{n+1}^2. \tag{5.7}$$

We apply formulae (4.7) to $-\tilde{E}_n$. The pieces are listed in Tables 1 and 2.

i	$\mathcal{L}^{(i+1)}_{u_n}(-\tilde{E}_n)$	$(\mathrm{D}-\mathrm{I})^i(u_n\mathcal{L}^{(i+1)}_{u_n}(-\tilde{E}_n))$
0	$u_{n-1}v_{n-1}+u_{n+1}v_n$	$u_nu_{n-1}v_{n-1}+u_nu_{n+1}v_n$
1	$u_{n-1}v_{n-1}$	$u_{n+1}u_nv_n-u_nu_{n-1}v_{n-1}$

TABLE 1. Computation of $j_{1,n}(\mathbf{u}_n)(-\tilde{E}_n)$

i	$\mathcal{L}^{(i+1)}_{v_n}(-\tilde{E}_n)$	$(\mathrm{D}-\mathrm{I})^i(v_n\mathcal{L}^{(i+1)}_{v_n}(-\tilde{E}_n))$
0	$u_nu_{n+1}+2v_n$	$v_nu_nu_{n+1}+2v_n^2$

TABLE 2. Computation of $j_{2,n}(\mathbf{u}_n)(-\tilde{E}_n)$

Adding the terms in the right columns in Table 1 and Table 2,

$$j_{1,n}(\mathbf{u}_n)(-\tilde{E}_n) = 2u_nu_{n+1}v_n, \quad j_{2,n}(\mathbf{u}_n)(-\tilde{E}_n) = u_nu_{n+1}v_n + 2v_n^2. \tag{5.8}$$

Thus, the homotopy operator (4.6) gives

$$\begin{aligned} \tilde{J}_n &= \int_0^1 \left(j_{1,n}(\mathbf{u}_n)(-\tilde{E}_n)[\lambda\mathbf{u}_n] + j_{2,n}(\mathbf{u}_n)(-\tilde{E}_n)[\lambda\mathbf{u}_n]\right)\frac{d\lambda}{\lambda} \\ &= \int_0^1 (3\lambda^2u_nu_{n+1}v_n + 2\lambda v_n^2)\, d\lambda \\ &= u_nu_{n+1}v_n + v_n^2. \end{aligned} \tag{5.9}$$

After a backward shift, $J_n = \mathrm{D}^{-1}\tilde{J}_n$, we obtain the final result:

$$\rho_n = \frac{1}{3}\, u_n^3 + u_n(v_{n-1} + v_n), \quad J_n = u_{n-1}u_n v_{n-1} + v_{n-1}^2. \tag{5.10}$$

6. Algorithm to Compute Generalized Symmetries

As an example, we compute the symmetry (2.30) of rank $(3,4)$ for (2.4). The two steps of the algorithm [**GH99**] are similar to those in Section 5.

Step 1: Construct the form of the symmetry

Start by listing all monomials in u_n and v_n of ranks 3 and 4, or less:

$$\mathcal{K}_1 = \{u_n^3, u_n^2, u_n v_n, u_n, v_n\}, \quad \mathcal{K}_2 = \{u_n^4, u_n^3, u_n^2 v_n, u_n^2, u_n v_n, u_n, v_n^2, v_n\}.$$

As in Step 1 in Section 5, for each monomial in $\mathcal{K}_1$ and $\mathcal{K}_2$, introduce the necessary t-derivatives so that each term has rank 3 and 4, respectively. At the same time, use (2.4) to remove all $t-$derivatives. Doing so, based on $\mathcal{K}_1$ we obtain

$$\mathcal{L}_1 = \{u_n^3, u_{n-1}v_{n-1}, u_n v_{n-1}, u_n v_n, u_{n+1}v_n\}. \tag{6.1}$$

Similarly, based on $\mathcal{K}_2$, we get

$$\begin{aligned} \mathcal{L}_2 \;=\; & \{u_n^4, u_{n-1}^2 v_{n-1}, u_{n-1}u_n v_{n-1}, u_n^2 v_{n-1}, v_{n-2}v_{n-1}, v_{n-1}^2, u_n^2 v_n, \\ & u_n u_{n+1} v_n, u_{n+1}^2 v_n, v_{n-1}v_n, v_n^2, v_n v_{n+1}\}. \end{aligned} \tag{6.2}$$

In contrast to the strategy for densities, we do not introduce the main representatives, but linearly combine the monomials in $\mathcal{L}_1$ and $\mathcal{L}_2$ to get the form of the symmetry:

$$\begin{aligned} G_1 &= c_1\, u_n^3 + c_2\, u_{n-1}v_{n-1} + c_3\, u_n v_{n-1} + c_4\, u_n v_n + c_5\, u_{n+1}v_n, \\ G_2 &= c_6\, u_n^4 + c_7\, u_{n-1}^2 v_{n-1} + c_8\, u_{n-1}u_n v_{n-1} + c_9\, u_n^2 v_{n-1} + c_{10}\, v_{n-2}v_{n-1} \\ &\quad + c_{11}\, v_{n-1}^2 + c_{12}\, u_n^2 v_n + c_{13}\, u_n u_{n+1} v_n + c_{14}\, u_{n+1}^2 v_n + c_{15}\, v_{n-1}v_n \\ &\quad + c_{16}\, v_n^2 + c_{17}\, v_n v_{n+1}, \end{aligned} \tag{6.3}$$

with constant coefficients c_i.

Step 2: Determine the coefficients

To compute the coefficients c_i we require that (2.22) holds on solutions of (2.1). Compute $\mathrm{D}_t G_1$ and $\mathrm{D}_t G_2$ and use (2.4) to remove $\dot{u}_n, \dot{v}_n$, and their shifts. This gives the left hand sides of (2.22).

Use (2.26) to get the right hand sides of (2.22):

$$\begin{aligned} F_1{}'(\mathbf{u}_n)[\mathbf{G}] &= \mathrm{D}^{-1}G_2 - \mathrm{I}G_2, \\ F_2{}'(\mathbf{u}_n)[\mathbf{G}] &= v_n \mathrm{I}G_1 - v_n \mathrm{D}G_1 + (u_n - u_{n+1})\mathrm{I}G_2. \end{aligned} \tag{6.4}$$

Substitute (6.3) into (6.4) and equate the corresponding left and right hand sides. Since all monomials in u_n, v_n and their shifts are independent, one obtains the linear system that determines the coefficients c_i. The solution is

$$\begin{aligned} & c_1 = c_6 = c_7 = c_8 = c_9 = c_{10} = c_{11} = c_{13} = c_{16} = 0, \\ & c_2 = c_3 = -c_4 = -c_5 = c_{12} = -c_{14} = c_{15} = -c_{17}. \end{aligned} \tag{6.5}$$

With the choice $c_{17} = 1$, the symmetry (6.3) finally becomes

$$\begin{aligned} G_1 &= v_n(u_n + u_{n+1}) - v_{n-1}(u_{n-1} - u_n), \\ G_2 &= v_n(u_{n+1}^2 - u_n^2 + v_{n+1} - v_{n-1}). \end{aligned} \tag{6.6}$$

7. Algorithm to Compute Scalar Recursion Operators

In this section we show how to compute the recursion operator (2.36) of (2.4).

Again, we will use the concept of rank invariance to construct a candidate recursion operator. The defining equation (2.32) is then used to determine the coefficients.

We observe that (2.36) in expanded form naturally splits into two pieces:

$$\mathcal{R} = \mathcal{R}_0 + \mathcal{R}_1, \tag{7.1}$$

where $\mathcal{R}_0$ contains only terms with shift operators $\mathrm{D}^{-1}, \mathrm{I}$, and D, and $\mathcal{R}_1$ has terms involving $(\mathrm{D} - \mathrm{I})^{-1}$.

Step 1: Determine the rank of the recursion operator

In view of (2.31) and assuming that the symmetries are linked consecutively ($s = 1$), the recursion operator $\mathcal{R}$ has rank

$$R = \text{rank } \mathcal{R} = \text{rank } G^{(2)} - \text{rank } G^{(1)} = 3 - 2 = 1, \tag{7.2}$$

where we used (2.27) and (2.28). If the assumption turns out to be correct, then the recursion operator has rank 1. If the assumption were falls because symmetries are not linked consecutively, then R must be adjusted and the subsequent steps must be repeated. See [**BHS04**] for examples of PDEs for which that happens.

Step 2: Determine the form of the recursion operator

We split this into two sub-steps.

(i) Determine the form of $\mathcal{R}_0$

The candidate $\mathcal{R}_0$ is a sum of terms involving $\mathrm{D}^{-1}, \mathrm{I}$, and D. The coefficients of these terms are linear combinations of u_{n-1}, u_n, and u_{n+1}, so that all terms have the correct rank.

$$\begin{aligned} \mathcal{R}_0 &= (c_1 u_{n-1} + c_2 u_n + c_3 u_{n+1})\,\mathrm{D}^{-1} + (c_4 u_{n-1} + c_5 u_n + c_6 u_{n+1})\,\mathrm{I} \\ &\quad + (c_7 u_{n-1} + c_8 u_n + c_9 u_{n+1})\,\mathrm{D}, \end{aligned} \tag{7.3}$$

where the c_i are constant coefficients.

A few remarks are in place. First, in $\mathcal{R}_0$ we moved the operators $\mathrm{D}^{-1}, \mathrm{I}$, and D all the way to the right. Second, the maximum up-shift and down-shift operator that should be included can be determined by comparing two consecutive symmetries. Indeed, if the maximum up-shift in the first symmetry is u_{n+p}, and the maximum up-shift in the next symmetry is u_{n+p+r}, then $\mathcal{R}_0$ must have $\mathrm{D}, \mathrm{D}^2, \cdots, \mathrm{D}^r$. The same line of reasoning determines the minimum down-shift operator to be included. In our example, there is no need to include terms in $\mathrm{D}^{-2}, \mathrm{D}^2$, etc. Third, the coefficients of the operators can be restricted to linear combinations of the terms appearing in F. Hence, no terms in $u_{n\pm2}, u_{n\pm3}$ and so on occur in (7.3).

(ii) Determine the form of $\mathcal{R}_1$

As in the continuous case [**HG99**], $\mathcal{R}_1$ is a linear combination (with constant coefficients $\tilde{c}_{jk}$) of sums of all suitable products of symmetries and covariants (Fréchet derivatives of densities) sandwiching $(\mathrm{D} - \mathrm{I})^{-1}$, i.e.

$$\mathcal{R}_1 = \sum_j \sum_k \tilde{c}_{jk} G^{(j)} (\mathrm{D} - \mathrm{I})^{-1} \rho_n^{(k)\prime}. \tag{7.4}$$

For (2.3), $G^{(1)}$ in (2.27) and $\rho_n^{(0)} = \ln(u_n)$ in (2.15) are the only suitable pair. Indeed, using (2.24) we have $\rho_n^{(0)\prime} = \ln(u_n)' = \frac{1}{u_n}\mathrm{I}$, which has rank -1. Combined with $G^{(1)}$ of rank 2, we have a term of rank 1. Other combinations of symmetries and covariants would exceed rank 1. Therefore,

$$\mathcal{R}_1 = \tilde{c}_{10} u_n (u_{n+1} - u_{n-1})(\mathrm{D} - \mathrm{I})^{-1}(\frac{1}{u_n})\mathrm{I} \tag{7.5}$$

where $\tilde{c}_{10}$ a constant coefficient. Using (7.1) and renaming $\tilde{c}_{10}$ to c_{10},

$$\begin{aligned} \mathcal{R} \; = \; & (c_1 u_{n-1} + c_2 u_n + c_3 u_{n+1})\mathrm{D}^{-1} + (c_4 u_{n-1} + c_5 u_n + c_6 u_{n+1})\mathrm{I} \\ & + (c_7 u_{n-1} + c_8 u_n + c_9 u_{n+1})\mathrm{D} + c_{10} u_n (u_{n+1} - u_{n-1})(\mathrm{D} - \mathrm{I})^{-1}(\frac{1}{u_n})\mathrm{I} \end{aligned} \tag{7.6}$$

is the candidate recursion operator for (2.3).

Step 3: Determine the coefficients

Starting from (7.6), we use (2.34) with $F = u_n(u_{n+1} - u_{n-1})$ to compute $\mathcal{R}'[F]$. The partial results will not be shown due to length.

Using (2.23), we compute

$$F' = -u_n \mathrm{D}^{-1} + (u_{n+1} - u_{n-1})\mathrm{I} + u_n \mathrm{D}. \tag{7.7}$$

Then we compute $\mathcal{R} \circ F'$ and $F' \circ \mathcal{R}$. After substituting the pieces into (2.32) we simplify the resulting expression using rules such as

$$\begin{aligned} (\mathrm{D} - \mathrm{I})^{-1}\mathrm{D} &= \mathrm{D}(\mathrm{D} - \mathrm{I})^{-1} = \mathrm{I} + (\mathrm{D} - \mathrm{I})^{-1}, \\ (\mathrm{D} - \mathrm{I})^{-1}\mathrm{D}^{-1} &= \mathrm{D}^{-1}(\mathrm{D} - \mathrm{I})^{-1} = -\mathrm{D}^{-1} + (\mathrm{D} - \mathrm{I})^{-1}. \end{aligned} \tag{7.8}$$

We further simplify by recursively using formulas like

$$\begin{aligned} \mathrm{D}U(u_n)(\mathrm{D}-\mathrm{I})^{-1}V(u_n)\mathrm{I} &= U(u_{n+1})V(u_n)\mathrm{I} + U(u_{n+1})(\mathrm{D}-\mathrm{I})^{-1}V(u_n)\mathrm{I}, \\ (\mathrm{D}-\mathrm{I})^{-1}U(u_n)V(u_n)\mathrm{D} &= U(u_{n-1})V(u_{n-1})\mathrm{I} + (\mathrm{D}-\mathrm{I})^{-1}U(u_{n-1})V(u_{n-1})\mathrm{I}. \end{aligned} \tag{7.9}$$

Finally, we equate like terms to obtain a linear system for the c_i. Substituting the solution

$$c_1 = c_3 = c_4 = c_7 = c_9 = 0, \quad c_2 = c_5 = c_6 = c_8 = c_{10} = 1 \tag{7.10}$$

into (7.6) we obtain the final result

$$\mathcal{R} = u_n \mathrm{D}^{-1} + (u_n + u_{n+1})\mathrm{I} + u_n \mathrm{D} + u_n(u_{n+1} - u_{n-1})(\mathrm{D} - \mathrm{I})^{-1}\frac{1}{u_n}\mathrm{I}. \tag{7.11}$$

A straightforward computation confirms that $\mathcal{R}G^{(1)} = G^{(2)}$ with $G^{(1)}$ in (2.27) and $G^{(2)}$ in (2.28).

8. Algorithm to Compute Matrix Recursion Operators

We construct the recursion operator (2.37) for (2.4). Now all the terms in (2.32) are 2×2 matrix operators.

Step 1: Determine the rank of the recursion operator

The difference in rank of symmetries is again used to compute the rank of the elements of the recursion operator. Using (2.8), (2.29) and (2.30),

$$\operatorname{rank} \mathbf{G}^{(1)} = \begin{pmatrix} 2 \\ 3 \end{pmatrix}, \quad \operatorname{rank} \mathbf{G}^{(2)} = \begin{pmatrix} 3 \\ 4 \end{pmatrix}. \tag{8.1}$$

Assuming that $\mathcal{R}\,\mathbf{G}^{(1)} = \mathbf{G}^{(2)}$, we use the formula

$$\operatorname{rank} \mathcal{R}_{ij} = \operatorname{rank} G_i^{(k+1)} - \operatorname{rank} G_j^{(k)}, \tag{8.2}$$

to compute a rank matrix associated to the operator

$$\operatorname{rank}\mathcal{R} = \begin{pmatrix} 1 & 0 \\ 2 & 1 \end{pmatrix}. \tag{8.3}$$

Step 2: Determine the form of the recursion operator

As in the scalar case, we build a candidate $\mathcal{R}_0$:

$$\mathcal{R}_0 \;=\; \begin{pmatrix} (\mathcal{R}_0)_{11} & (\mathcal{R}_0)_{12} \\ (\mathcal{R}_0)_{21} & (\mathcal{R}_0)_{22} \end{pmatrix}, \tag{8.4}$$

with

$$\begin{aligned}
(\mathcal{R}_0)_{11} &= (c_1 u_n + c_2 u_{n+1})\,\mathrm{I}, \\
(\mathcal{R}_0)_{12} &= c_3 \mathrm{D}^{-1} + c_4 \mathrm{I}, \\
(\mathcal{R}_0)_{21} &= (c_5 u_n^2 + c_6 u_n u_{n+1} + c_7 u_{n+1}^2 + c_8 v_{n-1} + c_9 v_n)\,\mathrm{I} \\
&\quad + (c_{10} u_n^2 + c_{11} u_n u_{n+1} + c_{12} u_{n+1}^2 + c_{13} v_{n-1} + c_{14} v_n)\,\mathrm{D}, \\
(\mathcal{R}_0)_{22} &= (c_{15} u_n + c_{16} u_{n+1})\,\mathrm{I}.
\end{aligned}$$

Analogous to the scalar case, the elements of matrix $\mathcal{R}_1$ are linear combinations with constant coefficients of all suitable products of symmetries and covariants sandwiching $(\mathrm{D}-\mathrm{I})^{-1}$. Hence,

$$\sum_j \sum_k \tilde{c}_{jk}\, \mathbf{G}^{(j)} (\mathrm{D}-\mathrm{I})^{-1} \otimes \rho_n^{(k)'}, \tag{8.5}$$

where $\otimes$ denotes the matrix outer product, defined as

$$\begin{pmatrix} G_1^{(j)} \\ G_2^{(j)} \end{pmatrix} (\mathrm{D}-\mathrm{I})^{-1} \otimes \begin{pmatrix} \rho_{n,1}^{(k)'} & \rho_{n,2}^{(k)'} \end{pmatrix} = \begin{pmatrix} G_1^{(j)}(\mathrm{D}-\mathrm{I})^{-1}\rho_{n,1}^{(k)'} & G_1^{(j)}(\mathrm{D}-\mathrm{I})^{-1}\rho_{n,2}^{(k)'} \\ G_2^{(j)}(\mathrm{D}-\mathrm{I})^{-1}\rho_{n,1}^{(k)'} & G_2^{(j)}(\mathrm{D}-\mathrm{I})^{-1}\rho_{n,2}^{(k)'} \end{pmatrix}.$$

Only the pair $(\mathbf{G}^{(1)}, \rho_n^{(0)'})$ can be used, otherwise the ranks in (8.3) would be exceeded. Using (2.26) and (2.18) we compute

$$\rho_n^{(0)'} = \begin{pmatrix} 0 & \frac{1}{v_n}\mathrm{I} \end{pmatrix}, \tag{8.6}$$

Therefore, using (8.5) and renaming $\tilde{c}_{10}$ to c_{17},

$$\mathcal{R}_1 = \begin{pmatrix} 0 & c_{17}(v_{n-1} - v_n)(\mathrm{D}-\mathrm{I})^{-1}\frac{1}{v_n}\mathrm{I} \\ 0 & c_{17} v_n (u_n - u_{n+1})(\mathrm{D}-\mathrm{I})^{-1}\frac{1}{v_n}\mathrm{I} \end{pmatrix}. \tag{8.7}$$

Adding (8.4) and (8.7) we obtain

$$\mathcal{R} \;=\; \begin{pmatrix} \mathcal{R}_{11} & \mathcal{R}_{12} \\ \mathcal{R}_{21} & \mathcal{R}_{22} \end{pmatrix}, \tag{8.8}$$

with

$$\begin{aligned}
\mathcal{R}_{11} &= (c_1 u_n + c_2 u_{n+1})\,\mathrm{I} \\
\mathcal{R}_{12} &= c_3 \mathrm{D}^{-1} + c_4 \mathrm{I} + c_{17}(v_{n-1} - v_n)(\mathrm{D}-\mathrm{I})^{-1}\frac{1}{v_n}\mathrm{I}, \\
\mathcal{R}_{21} &= (c_5 u_n^2 + c_6 u_n u_{n+1} + c_7 u_{n+1}^2 + c_8 v_{n-1} + c_9 v_n)\,\mathrm{I} \\
&\quad + (c_{10} u_n^2 + c_{11} u_n u_{n+1} + c_{12} u_{n+1}^2 + c_{13} v_{n-1} + c_{14} v_n)\,\mathrm{D}, \\
\mathcal{R}_{22} &= (c_{15} u_n + c_{16} u_{n+1})\mathrm{I} + c_{17} v_n (u_n - u_{n+1})(\mathrm{D}-\mathrm{I})^{-1}\frac{1}{v_n}\mathrm{I}.
\end{aligned}$$

Step 3: Determine the coefficients

All the terms in (2.32) need to be computed. The strategy is similar to the scalar case, yet the computations are much more cumbersome. Omitting the details, the result is: $c_2 = c_5 = c_6 = c_7 = c_8 = c_{10} = c_{11} = c_{12} = c_{13} = c_{15} = 0$, $c_1 = c_3 = c_4 = c_9 = c_{14} = c_{16} = -1$, and $c_{17} = 1$. Substitution the constants into (8.8) gives

$$\mathcal{R} = \begin{pmatrix} -u_n\mathrm{I} & -\mathrm{D}^{-1} - \mathrm{I} + (v_{n-1} - v_n)(\mathrm{D} - \mathrm{I})^{-1}\frac{1}{v_n}\mathrm{I} \\ -v_n\mathrm{I} - v_n\mathrm{D} & -u_{n+1}\mathrm{I} + v_n(u_n - u_{n+1})(\mathrm{D} - \mathrm{I})^{-1}\frac{1}{v_n}\mathrm{I} \end{pmatrix}. \tag{8.9}$$

It is straightforward to verify that $\mathcal{R}G^{(1)} = G^{(2)}$ with $G^{(1)}$ in (2.29) and $G^{(2)}$ in (2.30).

9. More Examples

EXAMPLE 9.1. The modified Volterra lattice [**ASY00, HH03**],

$$\dot{u}_n = u_n^2(u_{n+1} - u_{n-1}), \tag{9.1}$$

has two non-polynomial densities $\rho_n^{(0)} = \frac{1}{u_n}$ and $\rho_n^{(1)} = \ln(u_n)$, and infinitely many polynomial densities. The first two symmetries,

$$G^{(1)} = u_n^2(u_{n+1} - u_{n-1}), \tag{9.2}$$

$$G^{(2)} = u_n^2 u_{n+1}^2(u_n + u_{n+2}) - u_{n-1}^2 u_n^2(u_{n-2} + u_n), \tag{9.3}$$

are linked by the recursion operator

$$\mathcal{R} = u_n^2\mathrm{D}^{-1} + 2u_n u_{n+1}\mathrm{I} + u_n^2\mathrm{D} + 2u_n^2(u_{n+1} - u_{n-1})(\mathrm{D} - \mathrm{I})^{-1}\frac{1}{u_n}\mathrm{I}. \tag{9.4}$$

EXAMPLE 9.2. The AL lattice (2.2) has infinitely many densities [**GH98**] and symmetries [**GH99**]. The recursion operator is of the form (8.8) with

$$\mathcal{R}_{11} = P_n\mathrm{D}^{-1} - u_n\Delta^{-1}v_{n+1}\mathrm{I} - u_{n-1}P_n\Delta^{-1}\frac{v_n}{P_n}\mathrm{I},$$

$$\mathcal{R}_{12} = -u_n u_{n-1}\mathrm{I} - u_n\Delta^{-1}u_{n-1}\mathrm{I} - u_{n-1}P_n\Delta^{-1}\frac{u_n}{P_n}\mathrm{I},$$

$$\mathcal{R}_{21} = v_n v_{n+1}\mathrm{I} + v_n\Delta^{-1}v_{n+1}\mathrm{I} + v_{n+1}P_n\Delta^{-1}\frac{v_n}{P_n}\mathrm{I},$$

$$\mathcal{R}_{22} = (u_n v_{n+1} + u_{n-1}v_n)\mathrm{I} + P_n\mathrm{D} + v_n\Delta^{-1}v_{n-1}\mathrm{I} + v_{n+1}P_n\Delta^{-1}\frac{u_n}{P_n}\mathrm{I}, \tag{9.5}$$

where $P_n = 1 + u_n v_n$ and $\Delta = \mathrm{D} - \mathrm{I}$.

This recursion operator has an inverse, which is quite exceptional. The investigation of the recursion operator structure of the AL lattice is work in progress.

Acknowledgements

M. Hickman and B. Deconinck are gratefully acknowledged for valuable discussions. D. Baldwin is thanked for proof reading the manuscript.

References

[AC91] M. J. Ablowitz and P. A. Clarkson, *Solitons, nonlinear evolution equations and inverse scattering*, London Math. Soc. Lect. Note Ser. **149**, Cambridge University Press, Cambridge, U. K., 1991.

[AL75] M. J. Ablowitz and J. F. Ladik, *Nonlinear differential-difference equations*, J. Math. Phys. **16** (1975), 598–603.

[ASY00] V. E. Adler, A. B. Shabat and R. I. Yamilov, *Symmetry approach to the integrability problem*, Theor. Math. Phys. **125** (2000), 1603–1661.

[BHS04] D. Baldwin, W. Hereman and J. Sayers, *Symbolic algorithms for the Painlevé test, special solutions and recursion operators for nonlinear PDEs*, present proceedings.

[GH97a] Ü. Göktaş and W. Hereman, *Symbolic computation of conserved densities for systems of nonlinear evolution equations*, J. Symb. Comp. **2**4 (1997), 591–621.

[GH97b] ______, Mathematica package *InvariantsSymmetries.m* is available since 1997 from http://www.mathsource.com/cgi-bin/msitem?0208-932.

[GH98] ______, *Computation of conserved densities for nonlinear lattices*, Physica D **1**23 (1998), 425–436.

[GH99] ______, *Algorithmic computation of higher-order symmetries for nonlinear evolution and lattice equations*, Adv. Comp. Math. **1**1 (1999), 55–80.

[HG99] W. Hereman and Ü. Göktaş, *Integrability Tests for Nonlinear Evolution Equations*. Computer Algebra Systems: A Practical Guide, (M. Wester, ed.), Wiley, New York, 1999, pp. 211–232.

[HGCM98] W. Hereman, Ü. Göktaş, M. D. Colagrosso and A. J. Miller, *Algorithmic integrability tests for nonlinear differential and lattice equations*, Comp. Phys. Comm. **1**15 (1998), 428–446.

[HH03] M. S. Hickman and W. A. Hereman, *Computation of densities and fluxes of nonlinear differential-difference equations*, Proc. Roy. Soc. Lond. A (2003), in press.

[HM02] P. E. Hydon and E. L. Mansfield, *A variational complex for difference equations*, Found. Comp. Math. (2003), in press.

[KvM75] M. Kac and P. van Moerbeke, *On an explicitly soluble system of nonlinear differential equations related to certain Toda lattices*, Adv. Math. **1**6 (1975), 160–169.

[MH02] E. L. Mansfield and P. E. Hydon, *On a variational complex for difference equations*. The Geometrical Study of Differential Equations. Proc. NSF-CBMS Conf. (J. A. Leslie and T. P. Robart, eds.), American Mathematical Society, Providence, Rhode Island, 2002, pp. 121–129.

[M04] E. L. Mansfield and R. Quispel, *Towards a variational complex for the finite element method*, present proceedings.

[OZF93] W. Oevel, H. Zhang and B. Fuchssteiner, *Mastersymmetries and multi-Hamiltonian formulations for some integrable lattice systems*, Prog. Theor. Phys. **8**1 (1989), 294–308.

[O93] P. J. Olver, *Applications of lie groups to differential equations*, 2nd. ed., Graduate Texts in Mathematics **1**07, Springer, New York, 1993.

[SW01a] J. A. Sanders and J. P. Wang, *Integrable systems and their recursion operators*, Nonlinear Anal. **4**7 (2001), 5213–5240.

[SW01b] ______, *On recursion operators*, Physica D **1**49 (2001), 1–10.

[T81] M. Toda, *Theory of nonlinear lattices*, Springer, Berlin, 1981.

[W98] J. P. Wang, *Symmetries and conservation laws of evolution equations*, Ph.D. Thesis, Thomas Stieltjes Institute for Mathematics, Amsterdam, 1998.

Department of Mathematical and Computer Sciences, Colorado School of Mines, Golden, Colorado 80401-1887

E-mail address: `whereman@mines.edu`

Department of Mathematics, Faculty of Sciences, Free University, De Boelelaan 1081a, 1081 HV Amsterdam, The Netherlands

E-mail address: `jansa@cs.vu.nl`

Physics Department 103-33, California Institute of Technology, Pasadena, California 91125

E-mail address: `jack@its.caltech.edu`

Department of Mathematics, Brock University, St. Catharines, Ontario, Canada L2S 3A1

E-mail address: `jwang@spartan.ac.BrockU.ca`