An Automated Confluence Proof for an Infinite Rewrite System Parametrized over an Integro-Differential Algebra

Loredana Tec[1⋆], Georg Regensburger[2], Markus Rosenkranz[3], and Bruno Buchberger[1]

[1] Research Institute for Symbolic Computation, Johannes Kepler University, 4032 Castle of Hagenberg, Austria

[2] Johann Radon Institute for Computational and Applied Mathematics, Austrian Academy of Sciences, Altenberger Str. 69, 4040 Linz, Austria

[3] School of Mathematics, Statistics and Actuarial Science, University of Kent, Canterbury CT2 7NF, United Kingdom

1 Introduction

In our symbolic approach to boundary problems for linear ordinary differential equations we use the algebra of *integro-differential operators* as an algebraic analogue of differential, integral and boundary operators (Section 2). They allow to express the problem statement (differential equation and boundary conditions) as well as the solution operator (an integral operator called "Green's operator"), and they are the basis for operations on boundary problems like solving and factoring [14, 17]. A survey of the implementation is given in [18].

The integro-differential operators are realized by a noetherian and confluent rewrite system [17]. From a ring-theoretic point of view, this rewrite system constitutes a basis for the ideal of relations among the fundamental operators, and confluence means we have a noncommutative *Gröbner basis* [3, 4, 2, 9]. However, since the relation ideal is infinitely generated in a polynomial ring with infinitely many indeterminates, none of the known implementations [13] is applicable.

This is why the *confluence proof* is somewhat subtle (Section 3). The generators for the relation ideal are parametrized over a given integro-differential algebra, and the reduction of S-polynomials must incorporate the computational laws of the latter. The automated proof in [15] has achieved this in an ad-hoc manner for the special case of what was called "analytic algebras" there. In our new proof, the computational laws of integro-differential algebras are internalized by using so-called integro-differential polynomials [16] in the formation of the S-polynomials. We also refer to [19] for a detailed presentation of the new automated proof and the corresponding integro-differential structures.

We use a prototype *implementation* of integro-differential polynomials and reduction rings, based on Theorema and available at www.theorema.org. The Theorema system was designed by B. Buchberger as an integrated environment

⋆ Recipient of a DOC-fFORTE-fellowship of the Austrian Academy of Sciences.

for proving, solving and computing in various domains [6]. Implemented on top of Mathematica, its core language is higher-order predicate logic and contains a natural programming language such that algorithms can be coded and verified in a unified formal frame, using the powerful tool of functors for building up a hierarchy of parametrized domains; for more details and references see [8].

2 Integro-Differential Polynomials and Operators

We need an algebraic structure having *differentiation along with integration*. In the following definition [17], one may think of the standard example $\mathcal{F} = C^\infty(\mathbb{R})$, where $\partial = {}'$ is the usual derivation and $\int$ the integral operator $f \mapsto \int_a^x f(\xi)\, d\xi$ for $a \in \mathbb{R}$. The section axiom corresponds to the Fundamental Theorem of Calculus, the differential Baxter axiom to Integration by Parts. Scalars are over a field K. For the similar notion of differential Rota-Baxter algebras, we refer to [10].

Definition 1. *An* integro-differential algebra $(\mathcal{F}, \partial, \int)$ *is a commutative differential K-algebra $(\mathcal{F}, \partial)$ with a K-linear section $\int$ of ∂, meaning $(\int f)' = f$, such that the differential Baxter axiom $(\int f')(\int g') + \int (fg)' = (\int f')g + f(\int g')$ holds.*

Let $(\mathcal{F}, \partial, \int)$ be an integro-differential algebra of "coefficients". Then the *integro-differential operators* $\mathcal{F}[\partial, \int]$, introduced in [17] as a generalization of the "Green's polynomials" of [15], are defined as the quotient—modulo the rewrite rules from the table below—of the noncommutative polynomial ring over K in the following indeterminates: ∂ and $\int$, the "functions" $f \in \mathcal{F}$, and the multiplicative "functionals" φ. The functions f range over a basis of $\mathcal{F}$; the multiplicative functionals (or characters) $\varphi\colon \mathcal{F} \to K$ are typically point evaluations, and they must include the *evaluation* $\mathtt{E} = 1 - \int \partial$, which is $\mathtt{E}(f) = f(a)$ in the above example. In the rewrite rules, we use f and g range over functions, φ and ψ over multiplicative functionals.

$fg \to f \cdot g$	$\partial f \to \partial \cdot f + f\partial$	$\int f \int \to (\int \cdot f) \int - \int (\int \cdot f)$
$\varphi\psi \to \psi$	$\partial\varphi \to 0$	$\int f \partial \to f - \int (\partial \cdot f) - (\mathtt{E} \cdot f)\, \mathtt{E}$
$\varphi f \to (\varphi \cdot f)\, \varphi$	$\partial \int \to 1$	$\int f \varphi \to (\int \cdot f)\, \varphi$

Theorem 1. *The above rewrite system is noetherian and confluent.*

As explained before, one may find an outline of a manual proof for this theorem in [17], but the purpose of the present paper is to sketch a new automated proof based on the algebra of *integro-differential polynomials*. The precise definition as an instance of the universal polynomial construction [12, 7, 1] is tedious [16], but the underlying intuition is perfectly clear since one just adjoins an indeterminate function u to the given integro-differential algebra $\mathcal{F}$. The integro-differential polynomials are an extension of the usual differential polynomials [11] and in analogy we denote them by $\mathcal{F}\{u\}$. A proof of the following theorem can be found in [19].

Theorem 2. *The integro-differential polynomials $\mathcal{F}\{u\}$ constitute an integro-differential algebra with an algorithmic canonical simplifier.*

Unlike the integro-differential operators, $\mathcal{F}\{u\}$ is thus a commutative integro-differential algebra, and its multiplication is realized by the so-called shuffle product. While the definition of the derivation is straightforward and similar to differential polynomials, the integral must be defined by a careful case distinction on the differential exponents [16, 19]. Note that integro-differential polynomials act as nonlinear differential and integral operators on $\mathcal{F}$. A typical integro-differential polynomial for $\mathcal{F} = K[x]$ is given by $4u(0)^4 u^2 \int u'^3 + \int (x^6 u u''^5 \int (x^2 e^{4x} u^3 u'^2 \int u^4))$. For computational purposes, we have implemented a *canonical simplifier*, identifying different expressions that denote the same integro-differential polynomial.

3 An Automated Confluence Proof

As announced in the Introduction, the integro-differential polynomials are used for proving the confluence of the above rewrite rules defining the relations for $\mathcal{F}[\partial, \int]$. Equivalently, we show that the noncommutative polynomials given by the difference between the left and right sides of the rules form a noncommutative Gröbner basis. For handling parametrized polynomial reduction and S-polynomials, we use a noncommutative adaption of *reduction rings*, i.e. rings with so-called reduction multipliers in the sense of [5]. As usual, we show that all S-polynomials reduce to zero.

Since the rewrite rules contain two generic functions f and g, one can view the corresponding *S-polynomials* as elements of $\widetilde{\mathcal{F}}[\partial, \int]$ with $\widetilde{\mathcal{F}} = \mathcal{F}\{u, v\}$. Here $\mathcal{F}\{u, v\} = (\mathcal{F}\{u\})\{v\}$ denotes the integro-differential polynomials in two indeterminates. More precisely, we reason as follows: If we know that an S-polynomial reduces to zero as such, the same is true after substituting the functions $f, g \in \mathcal{F}$ for u, v. Note the subtle shift between object and meta level when we use the instance of the rewrite system for the integro-differential operators $\widetilde{\mathcal{F}}[\partial, \int]$ over integro-differential polynomials for proving the confluence of the rewrite rules for integro-differential operators over arbitrary integro-differential algebras—this proof needs only rewriting not confluence! (Actually one should also treat the functionals φ, ψ in analogy to the functions f, g, but the former are much simpler than the latter.) We refer again to [19] for further details.

We can now use Theorema for checking whether an S-polynomial reduces to zero. All S-polynomials are generated algorithmically, but as a concrete example we check the self-overlap $\int u \int$ and $\int v \int$ of the Baxter rule.

```
TS_In[554]:= ReducePol[(("∫"·u⟨1⟩) "∫"·v⟨1⟩ "∫" - "∫" ("∫"·u⟨1⟩) v⟨1⟩ "∫") -
               ("∫" u⟨1⟩ ("∫"·v⟨1⟩) "∫" - "∫" u⟨1⟩ "∫" ("∫"·v⟨1⟩))]
TS_Out[554]=
      0
```

It turns out that there are 72 S-polynomials, and indeed all of them reduce to zero. Hence we conclude that the rewrite system for $\mathcal{F}[\partial, \int]$ is confluent.

References

1. E. Aichinger and G. F. Pilz. A survey on polynomials and polynomial and compatible functions. In *Proceedings of the Third International Algebra Conference*, pages 1–16, Dordrecht, 2003. Kluwer Acad. Publ.
2. G. M. Bergman. The diamond lemma for ring theory. *Adv. in Math.*, 29(2):178–218, 1978.
3. B. Buchberger. *An algorithm for finding the bases elements of the residue class ring modulo a zero dimensional polynomial ideal (German)*. PhD thesis, Univ. of Innsbruck, 1965. English translation *J. Symbolic Comput.*, 41(3-4):475–511, 2006.
4. B. Buchberger. Introduction to Gröbner bases. In B. Buchberger and F. Winkler, editors, *Gröbner bases and applications*, pages 3–31. Cambridge Univ. Press, 1998.
5. B. Buchberger. Gröbner rings and modules. In *Proceedings of SYNASC 2001*, pages 22–25, 2001.
6. B. Buchberger et al. Theorema: Towards computer-aided mathematical theory exploration. *J. Appl. Log.*, 4(4):359–652, 2006.
7. B. Buchberger and R. Loos. Algebraic simplification. In *Computer algebra*, pages 11–43. Springer, Vienna, 1983.
8. B. Buchberger, G. Regensburger, M. Rosenkranz, and L. Tec. General polynomial reduction with Theorema functors: Applications to integro-differential operators and polynomials. *ACM Commun. Comput. Algebra*, 42(3):135–137, 2008.
9. J. Bueso, J. Gómez-Torrecillas, and A. Verschoren. *Algorithmic methods in non-commutative algebra*. Kluwer Academic Publishers, Dordrecht, 2003.
10. L. Guo and W. Keigher. On differential Rota-Baxter algebras. *J. Pure Appl. Algebra*, 212(3):522–540, 2008.
11. E. Kolchin. *Differential algebra and algebraic groups*, volume 54 of *Pure and Applied Mathematics*. Academic Press, New York-London, 1973.
12. H. Lausch and W. Nöbauer. *Algebra of polynomials*. North-Holland Publishing Co., Amsterdam, 1973.
13. V. Levandovskyy. PLURAL, a non-commutative extension of SINGULAR: past, present and future. In *Mathematical software—ICMS 2006*, volume 4151 of *LNCS*, pages 144–157. Springer, Berlin, 2006.
14. G. Regensburger and M. Rosenkranz. An algebraic foundation for factoring linear boundary problems. *Ann. Mat. Pura Appl. (4)*, 188(1):123–151, 2009.
15. M. Rosenkranz. A new symbolic method for solving linear two-point boundary value problems on the level of operators. *J. Symbolic Comput.*, 39(2):171–199, 2005.
16. M. Rosenkranz and G. Regensburger. Integro-differential polynomials and operators. In *Proceedings of ISSAC '08*, pages 261–268, New York, 2008. ACM.
17. M. Rosenkranz and G. Regensburger. Solving and factoring boundary problems for linear ordinary differential equations in differential algebras. *J. Symbolic Comput.*, 43(8):515–544, 2008.
18. M. Rosenkranz, G. Regensburger, L. Tec, and B. Buchberger. A symbolic framework for operations on linear boundary problems. In *Proceedings of CASC '09*, volume 5743 of *LNCS*, pages 269–283, Berlin, 2009. Springer.
19. M. Rosenkranz, G. Regensburger, L. Tec, and B. Buchberger. Symbolic analysis for boundary problems: From rewriting to parametrized Gröbner bases. Technical Report 2010-05, RICAM, 2010.