SOLVING LINEAR BOUNDARY VALUE PROBLEMS VIA NON-COMMUTATIVE GRÖBNER BASES

Communicated by Robert Gilbert

MARKUS ROSENKRANZ, BRUNO BUCHBERGER
Research Institute for Symbolic Computation
Johannes Kepler University Linz
4040 Linz, Austria

HEINZ W. ENGL
Institute for Industrial Mathematics
Johannes Kepler University Linz
4040 Linz, Austria

AMS: 34B05, 34B27, 47A05

Abstract A new approach for symbolically solving linear boundary value problems is presented. Rather than using general-purpose tools for obtaining parametrized solutions of the underlying ODE and fitting them against the specified boundary conditions (which may be quite expensive), the problem is interpreted as an operator inversion problem in a suitable Banach space setting. Using the concept of the oblique Moore-Penrose inverse, it is possible to transform the inversion problem into a system of operator equations that can be attacked by virtue of non-commutative Gröbner bases. The resulting operator solution can be represented as an integral operator having the classical Green's function as its kernel. Although, at this stage of research, we cannot yet give an algorithmic formulation of the method and its domain of admissible inputs, we do believe that it has promising perspectives of automation and generalization; some of these perspectives are discussed.

KEY WORDS: Linear boundary value problems, Green's function, Moore-Penrose equations, symbolic solution.

(Received for Publication — — —)

1. Introduction

Sophus Lie said in 1894 what is nowadays folklore [13, p. 488]: "All branches of physics pose problems that end up in integrating differential equations", and similar things can be said about many other sciences. A great deal of these differential equations come in the form of *boundary value problems*, and it is this problem type that has inspired rich parts of functional analysis, as one can see nicely in the classic work of Hilbert-Courant [9].

It is therefore natural to ask about *symbolic methods* for boundary value problems (BVP). But quite in contrast to the rich arsenal of numerical algorithms for BVP, this corner of mathematics seems to be a bit neglected by the "symbolic world". Of course, there are some standard techniques available for various kinds of differential equations—ordinary and partial, linear and nonlinear [10, 23, 19]. At the first glance, one might think this is sufficient, since one can always solve the corresponding differential equation and adapt the free coefficients of the generic solution to fit the boundary conditions.

However, we are not only asking for the solution of one individual differential equation generated by fixing the inhomogeneity on its right-hand side; what we want is a generic expression that can be instantiated by all admissible right-hand sides for producing the corresponding solutions (see below). Besides this, the generic solution might have no closed form whereas its "adaption" to the given boundary conditions often does.

Therefore we propose a new approach that works on the BVP as a whole, representing both the differential equation and the boundary condition by operators on suitable Banach spaces. Such a *functional-analytic setting* is of course very familiar in abstract convergence analysis of numerical BVP algorithms, but interestingly it turns out to be equally useful for searching symbolic solutions via non-commutative Gröbner bases. The idea is that both the differential and the boundary operator are built up from some "atomic" operators and can thus be seen as non-commutative polynomials with the atomic operators as their indeterminates. For obtaining suitable polynomial equations, we use the powerful concept of the oblique Moore-Penrose inverse [21].

1.1. Problem Formulation

In this paper, we consider *ordinary* differential operators and *linear* BVP; see Section 4 for a discussion of possible extensions. Furthermore we will search for solutions over a *finite interval* $[a, b]$. Now let T be a linear differential operator of order n, so for $u \in C^n[a, b]$ we have

$$T\,u = c_0\,u^{(n)} + \ldots + c_{n-1}\,u' + c_n\,u,$$

where $c_0, \ldots, c_n$ are sufficiently smooth functions (for example, $c_j \in C^{n-j}[a, b]$ for each $j = 0, \ldots, n$) and c_0 is non-zero throughout $[a, b]$. We view T as a linear operator on the Banach space $(C[a, b], ||\cdot||_\infty)$ with dense domain of definition $\mathcal{D}(T) = C^n[a, b]$. The boundary operators $B_1, \ldots, B_n$ are defined on the same domain; for each $i = 1, \ldots, n$ we have

$$\begin{aligned} B_i\ u = {} & p_{i,0}\,u^{(n)}(a) + \ldots + p_{i,n-1}\,u'(a) + p_{i,n}\,u(a) \\ & + q_{i,0}\,u^{(n)}(b) + \ldots + q_{i,n-1}\,u'(b) + q_{i,n}\,u(b), \end{aligned}$$

where the coefficients $p_{i,j}, q_{i,j}$ are real numbers. Now the *boundary value problem induced by T and $B_1, \ldots, B_n$* is to find for each right-hand side $f \in C[a, b]$ a function $u \in C^n[a, b]$ such that

$$\begin{aligned} & T\ u = f, \\ & B_1\,u = \ldots = B_n\,u = 0. \end{aligned} \tag{1}$$

This BVP is actually inhomogeneous in the differential equation and homogeneous in the boundary conditions (*semi-inhomogeneous problem*). But we can always decompose a fully inhomogeneous problem into such a semi-inhomogeneous one and a rather trivial BVP with homogeneous differential equation and inhomogeneous boundary conditions (*semi-homogeneous problem*); see [25, p. 43] for an explanation. Furthermore, we will assume throughout the paper that the boundary conditions are such that they determine a unique solution u of (1) for all $f \in C[a, b]$.

1.2. Solution Strategy

We are searching for an operator G that takes the inhomogeneity f as input and produces the solution u of (1) as output. In fact, in those cases which we consider, it is well-known

that the operator G can be written as an integral operator with the so-called *Green's function* g as its kernel [8, p. 296] via

$$G\,f(x) = \int_a^b g(x,\xi)\,f(\xi)\,d\xi. \tag{2}$$

The desired solution operator G is obviously a *right inverse* of the given differential operator: $T\,G\,(f) = f$ and hence $T\,G = 1$. (For the sake of simplicity, we will use the symbol 1 for denoting various identity functions and operators.) Of course, there are many right inverses for T, but the boundary conditions $B_1 u = \ldots = B_n u = 0$ are supposed to single out the one we want. It should be noted that this viewpoint is different from the standard one, where the boundary conditions are used for specifying the domain of the differential operator; in this case, there is of course only one inverse.

So we want to find a right inverse that is normally not an inverse in the strict sense—this is where the concept of the *oblique Moore-Penrose inverse* enters the stage (see Subsection 2.1 for details): Given the operator T on the Banach space $C[a,b]$ together with arbitrary projectors P, Q onto its nullspace and range closure, the oblique Moore-Penrose inverse $T^{\dagger}_{P,Q}$ can be determined by the four well-known Moore-Penrose equations, which can be seen as four non-commutative polynomial equations in the indeterminates $T, T^{\dagger}, P, Q$. By choosing suitable projectors P, Q, it may be possible to enforce the boundary conditions, which has the consequence that $T^{\dagger} = G$. In general, the projectors will thus become polynomials in $B_1, \ldots, B_n$ and some extra operators describing their particular structure. In many cases, one will be able to express some or all of the boundary operators as well as the differential operator T in terms of these extra operators. So let $A_1, \ldots, A_m$ be those boundary and extra operators that are left over; we will call them *auxiliary operators* altogether. Substituting them in the Moore-Penrose equations, we will end up with an equation system

$$\underset{i=1,\ldots,4}{\forall} \mathcal{P}_i(G, A_1, \ldots, A_m) = 0, \tag{3}$$

where $\mathcal{P}_1, \ldots, \mathcal{P}_4$ are some non-commutative polynomials in the indicated indeterminates.

Our goal is to obtain a partial triangularization of this system, i.e. to find an equivalent system containing an equation of the form $G = \ldots$, where the right-hand side should not contain G. This means we want a term representation for the solution operator G: it should be described in terms of some elementary operators like integration and multiplication. For giving a complete specification, we must therefore decide which *elementary operators* $E_1, \ldots, E_k$ we want to allow in the solution term for G. Depending on this choice, the task of triangularizing the equation system may be easy, difficult or even impossible. This is one of the critical points in our approach that spoils its algorithmic nature at the moment (see Section 4 for a brief discussion of this topic): We must either be creative in finding "good" elementary operators or we need powerful structure theorems for warranting the completeness of certain basis operators.

Assuming we have established a suitable collection of elementary operators $E_1, \ldots, E_k$, we must still specify how they are related with the auxiliary operators $A_1, \ldots, A_m$ occurring in the Moore-Penrose equations, i.e. we need some polynomial equations that describe their interaction. For example, if E_1 is integration and A_1 is differentiation, the obvious relation between them is the Fundamental Theorem of Calculus. This step is the second half of the "creative" phase just described; both steps should be taken together. Having found enough *interaction equations*

$$\underset{i=1,\ldots,l}{\forall} \mathcal{Q}_i(A_1, \ldots, A_m, E_1, \ldots, E_k) = 0, \tag{4}$$

we can combine (3) and (4), looking at it as a well-known problem of computer algebra: Given the ideal J induced by the polynomials $\mathcal{P}_1, \ldots, \mathcal{P}_4, \mathcal{Q}_1, \ldots, \mathcal{Q}_l$, try to find a basis

for J containing a polyomial with leading term G; see Equation (18) in Subsection 2.1 for an example. This means that we can write the corresponding equation in the desired form $G = \mathcal{G}(A_1, \ldots, A_m, E_1, \ldots, E_k)$, where $\mathcal{G}$ is a polynomial in the indicated indeterminates. If we have chosen suitable operators $A_1, \ldots, A_m, E_1, \ldots, E_k$, we can interpret the solution operator G as the usual Green's operator and extract from it the classical Green's function g.

For finding the desired basis, we use the *method of Gröbner bases*, introduced by the second author in his PhD thesis [3]; see also the journal version [4] and a concise treatment in [6]. The advantage of Gröbner bases is that they do not only lead to the desired solution but also reveal useful information about the ideal structure. In this paper, however, we will not address these issues. For a modern survey on the theory of Gröbner bases and their applications, see [7] and the remarks at the end of this section.

1.3. Previous Research

The idea of using the *Moore-Penrose inverse for solving linear BVP* is not new. One can find a standard treatment of this subject in [22] and [25]. But what is new, to our knowledge, is the observation that by means of non-commutative Gröbner bases one can actually utilize the Moore-Penrose equations for obtaining symbolic solutions. There is an interesting paper [18] from the seventies that describes a different Moore-Penrose method for approaching linear BVP. It is based on the concept of adjoint operators and orthogonal projectors (as opposed to the oblique ones used in our method), but it does not make use of Gröbner bases. This approach seems to result into more complex computations, but it would be an interesting research topic to combine this approach with our method.

Non-commutative Gröbner bases have been applied to differential operators for several decades, see for example the survey article [24] about Gröbner bases and partial differential equations. However, most of the theory in this field is concerned with studying the structure of solutions, without giving explicit methods for constructing them (the situation becomes even worse when it comes to BVP). Besides this, Gröbner bases have been used for *simplifying complicated operator expressions* as they typically arise in control theory. This approach is described in the papers [15, 16, 26] of the San Diego group, which also served as the starting point for our own investigations. We used the software package developed by their group for the Gröbner-basis computations necessary in our examples; see Subsection 2.3 for details.

The difference between the problem considered here and the subject of simplification addressed by their group is of a fundamental nature. Applications of Gröbner bases—both in the commutative and in the non-commutative cases—come in *three main categories* [6]:

- *Confluent Rewriting*: A Gröbner basis induces a rewrite system for reducing polynomials. Using a suitable term ordering, this will sometimes lead to a drastically simpler optical appearance, which is very important for control theorists [26]. However, the essential point is that the reduced form is not only optically simpler but even canonical, due to the characteristic Church-Rosser property of Gröbner basis. This means that one can decide equality: Two polynomials are equal in the given ideal if and *only* if their reduced forms are identical.

- *Polynomial Equation Solving*: Using a term ordering of the lexicographic type, Gröbner bases enjoy the so-called elimination property. Basically this means that the equation system will be triangularized as much as possible so that it is easy to solve the resulting system. The elimination property also holds in the non-commutative case; see [2].

- *Syzygies*: The information contained in the reductions that transform a given set of polyomials into a Gröbner basis can be used to determine the complete solution module of a linear equation system over a polynomial ring.

Seen in this way, research in the San Diego group belongs to the first category whereas our own research belongs to the second. It might be worthwile to also carry out operator-theoretic investigations in fields pertaining to the third application category.

1.4. Structure of the Paper

The rest of the paper is structured as follows: In Section 2 we take a well-known linear BVP as a simple but interesting example for *walking through the whole procedure* outlined above. We carry out and explain all the steps in detail; those passages that are only interesting for the specialized reader are printed in small font. In Section 3 we briefly present some *more examples* demonstrating different boundary conditions and slightly more complicated differential equations. In Section 4 we conclude with some *reflections* about the methodology and the potential of automation and generalization.

2. A Detailed Computation

The following problem seems to be one of the classical examples that are most often used for introducing the concepts of ordinary linear BVP [25, p. 42, 13, p. 480]. It can be interpreted as describing *one-dimensional steady heat conduction in a homogeneous rod*. We use it here as an instructive demonstration object for presenting our method in detail.

Given $f \in C[0,1]$,
find $u \in C^2[0,1]$
such that $u'' = f,\ u(0) = u(1) = 0$.

So the general problem described above is now given the simple instantiation $[a,b] = [0,1]$, $n = 2$, $T = D^2$, $B_1 = L$, $B_2 = R$. Here D^2 denotes the iterated differentiation operator on the Banach space $(C[0,1], \|\cdot\|_\infty)$; it has the subset $C^2[0,1]$ as its dense domain of definition. The left and right boundary operators L, R are defined in the obvious way: For each $u \in C[0,1]$, we have $L\ u = u(0)$ and $R\ u = u(1)$. Using this notation, we can rewrite the problem into the *operator form*:

$$\begin{aligned} D^2 u &= f, \\ L\,u = R\,u &= 0. \end{aligned}$$

As described above, we will interpret this now as an *inversion problem* in the following sense: Find a right inverse G of the operator D^2 such that the boundary conditions are also fulfilled, that is

$$\begin{aligned} D^2 G &= 1, \\ L\,G = R\,G &= 0. \end{aligned} \tag{5}$$

Note that from now on, we will use a slightly different interpretation for L, R, and 0: They are conceived as operators $C[0,1] \to C[0,1]$ rather than functionals $C[0,1] \to \mathbb{R}$, denoting the corresponding multiplication operators—multiplication by the left boundary value, by the right boundary value, by zero. So for any $u, v \in C[0,1]$ and $x \in [0,1]$, we have $((L\ u)v)(x) = u(0)v(x)$, $((R\ u)v)(x) = u(1)v(x)$, and $((0\ u)v)(x) = 0v(x) = 0$. This is necessary because we will end up with polynomials over the operator algebra on $C[0,1]$ having L, R as indeterminates and 0 as the zero polynomial.

2.1. The Moore-Penrose Equations

Our plan is to obtain the desired right inverse G as a suitable Moore-Penrose inverse. So let us recall the *Moore-Penrose equations* [12, p. 567]. For a densely defined closed operator $T : X \to Y$ between two Banach spaces X, Y such that both the nullspace $N := \mathcal{N}(T)$ and the closure of the range $R := \mathcal{R}(T)$ are continuously projectable, the *oblique Moore-Penrose inverse* $T^\dagger$ is determined by choosing projectors P and Q onto N and $\overline{R}$, respectively. If we abbreviate the corresponding topological complements by $M := (1-P)_* X$ and $S := (1-Q)_* Y$, the Moore-Penrose inverse $T^\dagger$ has domain $\mathcal{D}(T^\dagger) = R \oplus S$ and is characterized uniquely by the so-called Moore-Penrose equations

$$\begin{aligned} &T\,T^\dagger\,T = T, \\ &T^\dagger\,T\,T^\dagger = T^\dagger, \\ &T^\dagger\,T = (I-P)|_{\mathcal{D}(T)}, \\ &T\,T^\dagger = Q|_{\mathcal{D}(T^\dagger)}. \end{aligned} \tag{6}$$

Nullspace and *range* of $T^\dagger$ are given by [21, p. 33]

$$\begin{aligned} &\mathcal{N}(T^\dagger) = S, \\ &\mathcal{R}(T^\dagger) = M \cap \mathcal{D}(T). \end{aligned} \tag{7}$$

Furthermore,

$$T^\dagger \text{ is bounded} \Leftrightarrow R \text{ is closed;} \tag{8}$$

in this case we have of course $\mathcal{D}(T^\dagger) = Y$.

For $T = D^2$ and $X = Y = (C[0,1], ||\cdot||_\infty)$, the first conditions are obviously fulfilled, as T is indeed a densly defined and closed operator on the Banach space $(C[0,1], ||\cdot||_\infty)$. We will denote its Moore-Penrose inverse by G. The next step is to choose *suitable projectors* P and Q.

One could think that it is better to take $X = C^2[0,1]$ with its canonical norm: Then the operator D^2 would be defined everywhere, and it would even be continuous and not only closed; the operator P defined below would still be continuous. But the drawback of this setting is that $X = C^2[0,1]$ is not the "natural" domain for the operator D^2 as it is usually used in Sturm-Liouville theory. If the elements of X are obtained from measurements, the data error is typically estimated in the norm of $C[0,1]$, whereas it would not be reasonable to ask for estimates in the norm of $C^2[0,1]$, since this would involve the instable process of differentiating data [11]!

Let us start with the projector Q onto

$$\overline{R} = \overline{\mathcal{R}(D^2)} = \overline{C[0,1]} = C[0,1].$$

Since the *range* is the full space, the projector Q is trivial; the identity $1 : C[0,1] \to C[0,1]$ is the only possible choice,

$$Q = 1. \tag{9}$$

Furthermore, we see that the range is closed and so, by (8), G is bounded and $\mathcal{D}(G) = C[0,1]$.

Observe that the projector Q will always be the identity in the method described here. This reflects the fact that $T^\dagger = G$ is constructed as a right inverse, $T\,T^\dagger = 1$, so the projector Q is always 1 in the fourth equation of (6). The projector P, on the other hand, will always be nontrivial as long as we are dealing with a differential operator T, since it must have a nontrivial nullspace. In other words: The operator T is always surjective and non-injective.

The projector P maps $C[0,1]$ onto the *nullspace*

$$N = \mathcal{N}(D^2) = \{x \mapsto \alpha\ x + \beta \mid \alpha, \beta \in \mathbb{R}\},$$

so choosing P amounts to specifying for each $u \in C[0,1]$ real numbers α, β such that $(P\,u)(x) = \alpha\,x + \beta$ for all $x \in [0,1]$. We will use this freedom to ensure the boundary conditions by means of (7),

$$\mathcal{R}(G) = ((1-P)_*\,C[0,1]) \cap C^2[0,1] = (1-P)_*\,C^2[0,1]$$

We want to have $L\,u = 0$, $R\,u = 0$ for all $u \in \mathcal{R}(G)$, i.e. the counter-projections $(1-P)\,v = v - P\,v$ should vanish at the boundary for all $v \in C^2[0,1]$. Hence we must construct P such that $P\,v$ coincides with v for all $v \in C^2[0,1]$, and we may as well stipulate this condition for all $v \in C[0,1]$. Now this is a trivial *interpolation problem*: Given a function $v \in C[0,1]$, find a linear function $P\,v$ that agrees with v at the grid points $0, 1 \in [0,1]$. A short calculation leads to

$$(P\,u)(x) = (1-x)\,u(0) + x\,u(1). \tag{10}$$

It can be verified that P is continuous so that we can use the facts stated above about the Moore-Penrose equations (6). For applying the Gröbner bases method, we need a purely *operator-theoretic specification* of P. Defining the operator X for multiplication with the independent variable,

$$(X\,u)(x) = x\,u(x)$$

for all $u \in C[0,1]$ and $x \in [0,1]$, we can immediately rewrite (10) to

$$P = (1-X)\,L + X\,R. \tag{11}$$

Using (9) and (11), we can now write out the *concrete Moore-Penrose equations* (6) for the special case needed in our example:

$$\begin{aligned} &D^2\,G\,D^2 = D^2 \\ &G\,D^2\,G = G \\ &G\,D^2 = 1 - (1-X)\,L - X\,R \\ &D^2\,G = 1 \end{aligned} \tag{12}$$

We can see that they form indeed a *system of polynomial equations*, having the desired Green's operator G and the auxiliary operators (named $A_1, \ldots, A_m$ in the introduction) D, X, L, R as indeterminates. The only thing missing now are the elementary operators (named $E_1, \ldots, E_k$ in the introduction) that we want to allow in the solution term, together with suitable relations describing their interaction with the auxiliary operators.

The interpretation of (12) as a polynomial system involves a certain loss of information; it is a kind of projection that abolishes the topology and keeps only the algebraic structure. The problem is this: Being partial functions on $C[0,1]$, operators such as D cannot be composed arbitrarily with other operators; viewing the operators as indeterminates, however, one can build every composition by a suitable monomial. This has the consequence that the Gröbner-basis procedure will produce many useless S-polynomials that could be discarded by a simple domain check. In the established framework of Gröbner bases, however, this seems to be impossible; see Section 4 for an alternative approach.

The determination of the projector P is a *completely mechanical* task that could (and should!) be automated easily: In general, it involves solving a linear homogeneous differential equation, in this case just $u'' = 0$, and a linear interpolation problem induced by the boundary conditions. Hence the whole procedure of deriving the problem-specific Moore-Penrose equations as in (12) can be programmed in a computer algebra system.

2.2. The Interaction Equations

Now we come to the "creative" step of our approach (see Section 4 for a brief discussion about the potential of automation). It is clear that the operators D, X, L, R will not be sufficient for expressing the solution term for the Green's operator G. Since we would like to have an integral representation for G, having the corresponding Green's function g as its kernel, we must obvioulsy take the *antiderivative operator* A as one elementary operator. It is defined in the obvious way as

$$(A\,u)(x) = \int_0^x u(\xi)\,d\xi$$

for all $u \in C[0,1]$ and $x \in [0,1]$. What other elementary operators might be needed? In view of the duality in the boundary operators L, R, we may have the idea of adding the operator B *adjoint to the antiderivative operator* A. Whereas the operator A integrates *from* the left boundary, the operator B integrates *to* the right boundary, so it is defined as

$$(B\,u)(x) = \int_x^1 u(\xi)\,d\xi,$$

again for all $u \in C[0,1]$ and $x \in [0,1]$. Having A and B as elementary operators along with the auxiliary operators D, X, L, R, we hope that we can indeed express the solution operator G in the desired way.

The operators D, X, L, R, A, B will be meaningless unless we describe their *essential interactions*. Let us start with the interactions $D \leftrightarrow X$: Their atomic interaction is by composition. Whereas $X\,D$ cannot be simplfied anymore, the converse composition $D\,X$ can be rewritten as $X\,D + 1$ by the *product rule of differentiation*. All other polynomials in D and X can be expanded so that we can ultimately apply this atomic interaction. Hence we state as the first interaction equation

$$D\,X = X\,D + 1. \tag{13}$$

Another well-known relation for D is the *Fundamental Theorem of Calculus*; it describes interactions $D \leftrightarrow A, B$. For each of A, B, there are two interactions, namely

$$\begin{aligned} D\,A &= 1, \\ A\,D &= 1 - L, \\ D\,B &= -1, \\ B\,D &= R - 1. \end{aligned} \tag{14}$$

traditionally associated with the indefinite and definite integral, respectively. In this case, the interactions also involve the boundary operators L, R, so their relations to D, A, B are already covered. The only interaction still missing is $X \leftrightarrow L, R$. A short reflection leads to the *last two relations*

$$\begin{aligned} R\,X &= R, \\ L\,X &= 0. \end{aligned} \tag{15}$$

As mentioned before, at the moment there is no systematic procedure that would produce all of the necessary interactions. So there is no way to check whether we have *excluded a relevant equation*. But at least there is a way to check whether we have *included an irrelevant equation*: Any equation that is redundant with respect to the ones previously compiled can be recognized by reducing the corresponding polynomial modulo the ideal generated by the polynomials of the previous equations: it is redundant if and only if the reduction yiels zero. The computer algebra package *NCAlgebra,* used below for solving the ultimate system of equations, offers some functions for carrying out these reductions. If we do this for the above equations (13), (14), (15), we will see that there is no redundancy.

Combining equations (13), (14), (15), we end up with the *system of interaction equations*

$$
\begin{aligned}
&D\,X = X\,D + 1,\\
&D\,A = 1,\\
&A\,D = 1 - L,\\
&D\,B = -1,\\
&B\,D = R - 1,\\
&R\,X = R,\\
&L\,X = 0.
\end{aligned}
\tag{16}
$$

2.3. Solving the Equations

At this point, we have assembled the *complete polynomial equation system*, consisting of the polynomials $\mathcal{P}_1, \ldots, \mathcal{P}_4$ in the Moore-Penrose equations (12) and the polynomials $\mathcal{Q}_1, \ldots, \mathcal{Q}_7$ of the interaction equations (16), namely

$$
\begin{aligned}
&D^2 G\,D^2 - D^2,\\
&G\,D^2 G - G,\\
&G\,D^2 - 1 + (1 - X)L + X\,R,\\
&D^2 G - 1,\\
&D\,A - 1,\\
&A\,D - 1 + L,\\
&D\,B + 1,\\
&B\,D - R + 1,\\
&D\,X - 1 - X\,D,\\
&R\,X - R,\\
&L\,X.
\end{aligned}
\tag{17}
$$

Our goal is to solve the system (12), (16) for G, i.e. we want to find the elimination ideal of the polynomials (17) with respect to G. For this we will use the following *multigraded lexicographic term ordering*:

$$D < R < L < X < A < B \ll G \tag{18}$$

For solving polynomial equation systems, one typically uses lexicographic term orderings. However, their computational complexity is rather large. In our case, we do not need a complete triangularization of the equation system; it is enough to isolate the indeterminate G. Hence it is most efficient to use a multigraded ordering with G in the "highest" block. The ordering of the operators within the lower block is chosen to reflect their relative simplicity.

For computing the desired elimination ideal, we use the system *NCAlgebra* [14], a Mathematica package for doing non-commutative computer algebra, written by J. William Helton (Mathematics Department of the University of California, San Diego, California) and Robert L. Miller (General Atomic Corporation, La Jolla, California). It includes support for non-commutative Gröbner bases, also described in the papers [15, 16, 26].

First we *initialize* the system, designating D, R, L, X, A, B, G as non-commutative indeterminates and specifying the term ordering (18):

```
Get[ToFileName[{/, zvol, ncalgebra, NC}, SetNCDir.m]]
Get[ToFileName[{/, zvol, ncalgebra, NC, NCAlgebra}, NCAlgebra.m]]
Get[ToFileName[{/, zvol, ncalgebra, NC, NCGB, MmaSource}, NCGB.m]]
```

```
SetNonCommutative[D, R, L, X, A, B, G];

ClearMonomialOrderAll[]; ClearMonomialOrder[];
SetMonomialOrder[{D, R, L, X, A, B}, {G}]
```

Next we *enter the polynomials* (17) in the format of *NCAlgebra* (the operator $**$ denotes non-commutative multiplication, i.e. operator composition in our case):

```
initial = {D ** D ** G ** D ** D - D ** D,
  G ** D ** D ** G - G,
  G ** D ** D - 1 + (1 - X) ** L + X ** R,
  D ** D ** G - 1, D ** A - 1, A ** D - 1 + L, D ** B + 1,
  B ** D - R + 1, D ** X - 1 - X ** D, R ** X - R, L ** X}
```

Finally, we can invoke the command for *computing the desired elimination* of G:

```
final = NCMakeGB[initial, 2] // ColumnForm

-1 + D ** A
-1 + L + A ** D
1 + D ** B
1 - R + B ** D
1 - D ** X + X ** D
...
G + A ** X + X ** B - X ** A ** X - X ** B ** X
-R + L ** R
-R + D ** X ** R
-A - B + B ** A + B ** X + X ** A
...
```

The second argument in the command *NCMakeGB*, which is set to 2, specifies the number of search cycles used for producing S-polynomials. Unlike in the commutative case, Gröbner bases for non-commutative polynomial ideals may be infinite; hence one must prescribe a certain search depth for avoiding infinite runs. The consequence is that we obtain only a partial Gröbner bases (which is not even reduced), but this does not spoil our goal of isolating the indeterminate G.

The system has produced 42 polynomials, most of which are left out above, as they are not interesting for our present purposes; e.g. some of them express integration rules for polynomials such as $2x^2 + 3x$. The only important thing is that there is *precisely one polynomial involving the solution operator* G, and in this polynomial G does indeed occur isolated, just as desired. Writing the result in the usual format, we arrive at

$$G = X\,A\,X - A\,X + X\,B\,X - X\,B. \tag{19}$$

The steps carried out in this subsection are *clearly mechanical*, since we need no human support for the computation of the partial Gröbner basis once the polynomial equation system is fixed. However, this computation may not always lead to the desired result within a reasonable iteration bound. The problem is the exponential growth rate of the polynomials produced (it seems that *NCAlgebra* does not minimize the partial bases between iterations). In a more complicated example, one might need a portion of good luck to find a suitable term ordering and an iteration depth just high enough for producing the desired elimination without wasting too much time with producing many useless polynomials.

2.4. Interpreting the Result

For interpreting the result (19) in terms of a *Green's function*, we collect the terms belonging to the same integration operator,

$$G = (X-1)A\,X + X\,B(X-1).$$

Now we simply unfold the definion of the operators A, B, X as they are applied to a function $f \in C[0,1]$, evaluated at a point $x \in [0,1]$, giving

$$\begin{aligned}(G\,f)(x) &= (x-1)\int_0^x \xi\,f(\xi)\,d\xi + x\int_x^1 (\xi-1)\,f(\xi)\,d\xi \\ &= \int_0^x (x-1)\,\xi\,f(\xi)\,d\xi + \int_x^1 x\,(\xi-1)\,f(\xi)\,d\xi.\end{aligned}$$

Writing this as a single integral such that (2) holds, we must pack both integrands into one function, which we define by the corresponding *case distinction* as

$$g(x,\xi) = \begin{cases} (x-1)\xi & \text{if} \quad 0 \le \xi \le x \le 1, \\ x(\xi-1) & \text{if} \quad 0 \le x \le \xi \le 1. \end{cases}$$

It should be noted that the procedure of transforming G in (19) into the corresponding Green's function g is again *mechanical*: the monomials with A go into the first branch of g, those with B into the second branch; the occurrences of X before the integration operators become x, those after the integration operators become ξ. This mechanism also works for all the examples considered in the next section, where there are multiple occurrences of X-like operators like E, F in the example about damped oscillations.

3. Other Examples

In the previous section, we have treated a suitable sample problem in great detail. We will now present a couple of other examples, indicating only the main steps in the solution procedure; the computational details are similar to the sample problem.

3.1. Damped Oscillations

For a slightly more complicated problem, we take Example 2 in Kralle's book [17, p. 109]; the differential operator of this BVP has *damped oscillations* as its eigenfunctions [17, p. 107]. Stated in our terminology, the problem reads is

$$\begin{aligned} -\frac{(e^{2x}u(x)')' + e^{2x}u(x)}{e^{2x}} &= f(x), \\ u(0) = u(\pi) &= 0. \end{aligned} \tag{20}$$

Here x is assumed to range over the interval $[0,\pi]$. The notation T' is an abbreviation for $\frac{d}{d\,x}T$, where the differentiation quantifier $\frac{d}{d\,x}$ operates on the term T. For rewriting (20) as an operator equation, we introduce the some *auxiliary operators*. For all $u \in C[0,\pi], v \in C^1[0,\pi]$, and $x \in [0,\pi]$, we define

$$\begin{aligned}(D\,v)(x) &= v(x)', \\ (E\,u)(x) &= e^x\,u(x), \\ (F\,u)(x) &= e^{-x}\,u(x), \\ (L\,u)(x) &= u(0), \\ (R\,u)(x) &= u(\pi).\end{aligned}$$

Using these operators, the BVP (20) can be restated in *operator-theoretic form* as

$$\begin{aligned} -(F^2\,D\,E^2\,D+1)\,G &= 1,\\ L\,G = R\,G &= 0. \end{aligned}$$

For computing the non-trival projector P, one needs the *nullspace* of the operator $F^2\,D\,E^2\,D+1$: Whereas this was trivial for the operator D^2 of the previous examples, we must now solve a simple homogeneous differential equation. Using standard methods, one arrives at

$$\mathcal{N}(F^2\,D\,E^2\,D+1) = \{x \mapsto \alpha\,xe^{-x} + \beta\,e^{-x} \mid \alpha,\beta \in \mathbb{R}\}.$$

Going through the same procedure as with the operator D^2, one finds

$$P = \frac{e^\pi}{\pi}\,X\,F\,R - \frac{1}{\pi}\,X\,F\,L + F\,L, \tag{21}$$

where the operator X is defined as before.

As *elementary operators*, it is still sufficient to take A and B as in the D^2 example. As interactions, we need of course all those that we had before. The corresponding equations remain the same except for the interaction $X \leftrightarrow R$, which is now $R\,X = \pi\,R$.

In addition to the old interactions, we must specify *suitable relations for the new operators* E, F. First of all, the interactions $E \leftrightarrow F$ among themselves, namely the trivial fact that they are inverses of each other (hence it will be sufficient to describe the interactions with E alone in what follows), so $E\,F = F\,E = 1$. Then we must describe the interaction with differentiation $D \leftrightarrow E$. This amounts to a version of the product rule of differentiation, namely $D\,E = E\,D + E$. We also need the interactions with the boundary operators $L, R \leftrightarrow E$. One immediately sees that they are given by these two equations $L\,E = L$ and $R\,E = e^\pi\,R$. Finally, there is a trivial interaction $X \leftrightarrow E$, namely their commutativity, $E\,X = X\,E$.

Collecting all these interaction equations and combining them with the Moore-Penrose equations after substituting (21) for the nullspace projector and the identity for the range projector, one can solve the resulting polynomial system for G, using for example the *term ordering*

$$D < R < L < X < E < F < A < B \ll G.$$

Carrying out the computation in *NCAlgebra*, one obtains with iteration depth 3 (after applying some tedious tricks for representing the "commuting variables" e, π) the *result*

$$\begin{aligned} G &= F\,A\,X\,E + X\,F\,B\,E - \frac{1}{\pi}\,X\,F\,A\,X\,E - \frac{1}{\pi}\,X\,F\,B\,X\,E\\ &= \frac{1}{\pi}\,(\pi - X)\,F\,A\,X\,E + \frac{1}{\pi}\,X\,F\,B\,(\pi - X)\,E. \end{aligned}$$

This time, the partial basis contains 164 polynomials, but there is still only one among them involving G, namely exactly the one corresponding to the solved equation above. Going through the usual translation procedure, one can write G as an integral operator with the *Green's function*

$$g(x,\xi) = \begin{cases} \frac{1}{\pi}\,(\pi - x)\,\xi\,e^{\xi - x} & \text{if} \quad 0 \le \xi \le x \le \pi,\\ \frac{1}{\pi}\,(\pi - \xi)\,x\,e^{\xi - x} & \text{if} \quad 0 \le x \le \xi \le \pi. \end{cases}$$

This agrees with [17, p. 110], which is just an equivalent formulation using an extra kernel $e^{2\xi}$ in the Green's operator.

3.2. Heat Conduction with Variable Heat Conductivity

Encouraged by the previous example, we will now attack a one-dimensional heat equation with a variable conductivity coefficient, which we try to handle in a similar manner as the exponential function occurring in the problem of damped oscillations. The example is taken from [25, p. 69], prescribing a Dirichlet condition for the left boundary and a Neumann condition for the right boundary via

$$\begin{aligned} -(\,k(x)\,u(x)'\,)' &= f(x), \\ u(0) = u'(1) &= 0. \end{aligned} \tag{22}$$

We introduce the *auxiliary operator* K for the multiplication induced by the thermal conductivity k, analogous to the E operator induced by $x \mapsto e^x$ before. Furthermore, we introduce a second auxiliary operator J for the multiplication induced by its reciprocal, again similar to the F operator induced by $x \mapsto e^{-x}$ before. So for all $u \in C[0,\pi]$ and $x \in [0,\pi]$, we define

$$\begin{aligned} (K\,u)(x) &= k(x)\,u(x), \\ (J\,u)(x) &= k(x)^{-1}\,u(x). \end{aligned}$$

All the other operators are as in the steady heat-conduction example of Section 2. Hence we can give the *operator-theoretic formulation*

$$\begin{aligned} -D\,K\,D\,G &= 1, \\ L\,G = R\,D\,G &= 0. \end{aligned}$$

For computing the non-trivial projector P, we have to determine the *nullspace* of the given differential operator, which is

$$\mathcal{N}(D\,K\,D) = \{x \mapsto \alpha \int_0^x k(t)^{-1}\,dt + \beta \mid \alpha,\beta \in \mathbb{R}\}.$$

Here we can observe a new phenomenon: the nullspace—and hence the nullspace projector—involves integration. But this should not be surprising, as we are trying to solve a boundary value problem that is parametrized by a coefficient function in its differential operator. Carrying on with the computation procedure outlined before, we obtain

$$P = k_1\,A\,J\,R\,D + L$$

after a simple calculation, where k_1 is the constant $k(1)$. This concludes the compilation of the concrete Moore-Penrose equations. Since we have not used the multiplication operator X up to now, we expect that it is redundant in this example. Looking at the three interaction equations that involve the indeterminate X, we can see that they are only needed if this indeterminate occurs somewhere in the first place.

All the other *interaction equations* from (16) are still necessary, and we will have to add some new ones similar to the ones added in the foregoing example: The interactions $K\,J = J\,K = 1$ express the fact that K and J are inverses of each other, and the interactions with the boundary operators are $L\,J = k_0^{-1}L$, $R\,J = k_1^{-1}\,R$ for the J operator and $L\,K = k_0\,L$, $R\,K = k_1\,R$ for the K operator; here k_0 is of course the constant $k(0)$. The product rule for K and J is not needed in this example.

If one starts the Gröbner basis computation with these data, one does get a "solution" for the *Green's operator*, but it is of a somewhat unusual form. Again, there is only one equation that involves G, namely $G - A\,J\,B = 0$. Hence we obtain the operator

$$(G\,f)(x) = \int_0^x k(t) \left(\int_t^1 f(s)\,ds \right) dt,$$

which does not have the form of a single integral operator with a Green's function as its kernel. However, a moment's thought reveals that we just have to add one more interaction equation, namely $A\,J\,B = \tilde{J}\,B + A\,\tilde{J}$, where $\tilde{J}$ is the multiplication operator induced by the function $\int_0^x k(t)^{-1}\,dt$. This is nothing else than integration by parts applied to the function in J; similar rules could added for $A\,J\,A$, $B\,J\,A$, $B\,J\,B$, plus four additional ones with K in place of J. In our present example, the additional rules are not needed, though they would do no harm; in a more general case, the systematic approach would be to add them all.

Of course, after adding this interaction, we immediately get the solution in the form $G = \tilde{J}\,B + A\,\tilde{J}$, from which we can routinely extract the classical Green's function

$$g(x,\xi) = \begin{cases} \int_0^\xi k(t)^{-1}\,dt & \text{if} \quad 0 \le \xi \le x \le \pi, \\ \int_0^x k(t)^{-1}\,dt & \text{if} \quad 0 \le x \le \xi \le \pi, \end{cases}$$

as it is given in [25, p. 70]. This example seems particularly interesting to us, since it indicates a way to handle symbolic parameters in the BVP and it also demonstrates how to handle typical problems that can arise when setting up the proper interaction equations.

3.3. Transverse Deflection

The method presented in this paper is not restricted to the classical setting of second-order Sturm-Liouville theory. For seeing this, we take a practically relevant fourth-order problem [25, p. 49], which describes the *transverse deflection* $u \in C^2[0,1]$ of a homogeneous beam with distributed transversal load $f \in C[0,1]$, simply supported at both ends via

$$\begin{aligned} &u^{(4)} = f, \\ &u(0) = u(1) = u''(0) = u''(1) = 0. \end{aligned}$$

Its *operator-theoretic formulation* is

$$\begin{aligned} &D^4\,G = 1, \\ &L\,G = R\,G = L\,D^2\,G = R\,D^2\,G = 0. \end{aligned} \tag{23}$$

Comparing this BVP with the simple heat-conduction problem considered in (5), we observe a strong *similarity*. In fact, the only difference is the order of the differential operator and the additional boundary conditions for u''. Therefore, we expect that we can use the same auxiliary and elementary functions as in solving (5). This expectation is indeed fulfilled.

Going through the same procedure as in the heat-condution example, the boundary conditions in (23) lead to the *nullspace projector*

$$P = \frac{1}{6}\,X^3(R\,D^2 - L\;D^2) + \frac{1}{2}\,X^2 L\,D^2 + \frac{1}{6}\,X(6\,R - 6\,L - 2\,L\,D^2 - R\,D^2) + L$$

Using this operator and the interaction equations (16) from the heat-conduction problem, we obtain a polynomial system that can be solved for G by using the term ordering $X < D < A < B < R < L \ll G$. The partial basis returned in iteration depth 3 consists of 67 polynomials, and exactly one polynomial among them contains the indeterminate G for the *Green's operator*. Written as an equation, this polynomial is

$$\begin{aligned} G = {} & \frac{1}{3}\,X\,A\,X - \frac{1}{6}\,A\,X^3 - \frac{1}{2}\,X^2\,A\,X + \frac{1}{6}\,X\,A\,X^3 + \frac{1}{6}\,X^3\,A\,X \\ & + \frac{1}{3}\,X\,B\,X - \frac{1}{2}\,X\,B\,X^2 - \frac{1}{6}\,X^3\,B + \frac{1}{6}\,X\,B\,X^3 + \frac{1}{6}\,X^3\,B\,X. \end{aligned}$$

As usual, we can immediately translate this expression to the more traditional formulation in terms of a *Green's function* g defined as

$$g(x,\xi) = \begin{cases} \frac{1}{3}\,x\,\xi - \frac{1}{6}\,\xi^3 - \frac{1}{2}\,x^2\xi + \frac{1}{6}\,x\,\xi^3 + \frac{1}{6}\,x^3\xi & \text{if} \quad 0 \le \xi \le x \le 1, \\ \frac{1}{3}\,x\,\xi - \frac{1}{2}\,x\,\xi^2 - \frac{1}{6}\,x^3 + \frac{1}{6}\,x\,\xi^3 + \frac{1}{6}\,x^3\xi & \text{if} \quad 0 \le x \le \xi \le \pi. \end{cases}$$

This is in full accordance with [25, p. 71], where the result was obtained by means of the *causal fundamental solution*

$$h(x,\xi) = \begin{cases} \frac{(x-\xi)^3}{6} & \text{if} \quad 0 \le \xi \le x \le 1, \\ 0 & \text{if} \quad 0 \le x \le \xi \le \pi. \end{cases}$$

such that the Green's function takes on the form

$$g(x,\xi) = h(x,\xi) + \frac{\xi(1-\xi)(2-\xi)}{6}\,x - \frac{1-\xi}{6}\,x^3.$$

4. Conclusion

We have presented a new *method for solving linear boundary value problems by symbolic techniques*. It proceeds by transforming the given differential equation and its boundary conditions into a system of polynomial equations that can be solved for the desired Green's operator via non-commutative Gröbner bases. Of course, one must specify those operators and properties that should be used for representing the solution term; using the traditional framework of integral operators, one obtains a solution in terms of the usual Green's function. We have demonstrated in several examples how to compile suitable interaction equations leading to such a Green's formulation of the solution. (Incidentally, we have also found other representations of the solution, typically involving multiple integrations as in Subsection 3.2. Though outside the framework of the traditional Green's functions, these representations may be of numerical interest due to their smoothness properties.)

Let us now briefly analyze the *current status of algorithmization* in this method. In a typical application, it will proceed through the following steps:

- *Derivation of the concrete Moore-Penrose equations*: The major task in this step is to determine the nullspace projector P, since we have seen that the range projector Q is always the identity. Substituting P, Q and the given differential operator T in (6) and renaming the Moore-Penrose inverse $T^\dagger$ into G, we obtain the concrete Moore-Penrose equations. The polynomial for P will contain various auxiliary operators $A_1, \ldots, A_m$, usually made up from parts of the differential and boundary operators.

- *Compilation of the interaction equations*: After selecting suitable elementary operators $E_1, \ldots, E_k$, we have to find sufficiently many equations describing the relations between the auxiliary operators $A_1, \ldots, A_m$ and the elementary operators $E_1, \ldots, E_k$.

- *Computation of a partial Gröbner basis*: The concrete Moore-Penrose equations are combined with the interaction equations and supplied to a non-commutative Gröbner basis system, using a term ordering that isolates the Green's operator G.

- *Extraction of the Green's function*: The Green's operator G obtained in the previous step is transformed into the corresponding Green's function g.

As pointed out in Section 2, for the first step and the last two steps, our method can be viewed as an algorithm relative to the solvability of the homogeneous differential equation. For the second step, some *ad-hoc inventions* are still necessary for each problem instance at hand. In particular, one has to provide suitable interaction equations for specifying the solution structure. Some experience in handling BVP should be sufficient for finding these equalities.

Note that, after having found some basic interaction equations, the question of *how and in which order* these equations should be applied is exhaustively answered by the method of Gröbner bases, due to their Church-Rosser property: the given system of equations is automatically "completed". In a manual calculation, one has to play around with many possible ways of combining equations, which may or may not lead to success. In this sense, an essential portion of the usual "tricks" occurring in manual calculations is covered by our method; the remaining "tricks" can be seen in the interaction equations.

We believe that our method can cover various interesting classes of BVP, which we plan to explore in forthcoming papers (including some of the generalizations discussed below). In an ideal situation—presumably hard to achieve—one might approach a systematic search of elementary operators and interaction equations in a manner similar to the *structure theorems* of Liouville theory, which are used for indefinite integration and ordinary differential equations [10, p. 186]. Fixing certain algebraic input domains (e.g. the elementary transcendental functions) for the coefficients of the differential and boundary operators, one might be able to isolate a suitable "Green's domain" $\mathcal{G}$ such that the Green's function g will always be in $\mathcal{G}$. We think that such an expectation is realistic because it is well-known that one can express g in terms of solutions of certain initial value problems. (Note also that we do not claim that the solution of the BVP itself, namely $G\,f$ for a given right-hand side f, should have any algebraically simple form.)

Having found a Green's domain $\mathcal{G}$, it is probably not difficult to isolate appropriate *elementary operators* along with their *interaction equations*. Some of these operators might be multiplication operators induced by functions from the input domains and $\mathcal{G}$, similar to F and G in Subsection 3.1. It should also be observed that most of the interaction equations considered in this paper would come out quite naturally when the elementary operators are introduced in systematic exploration cycles as described in [5].

Finally, let us propose further lines of future research. The problems considered in this paper have some obvious generalizations. Increasing the number of independent variables leads us to BVP for *partial differential equations* like the well-known Dirichlet problem for $\mathrm{div}(\mathrm{a}\,\mathrm{grad}\,\mathrm{u}) = \mathrm{f}$. These equations will typically involve differential operators $\mathrm{div}, \mathrm{grad}, \mathrm{rot}, \ldots$ from vector analysis: We can either regard them as operators in their own right or assemble them from the partial differential operators $\partial_x, \partial_y, \partial_z$. The general methodology presented in this paper should be applicable in both cases; identities like the Divergence Theorem of Gauss will take over the role of the Fundamental Theorem of Calculus.

Passing over to *non-linear problems* is a much bigger challenge. In this case, compound operators may not be expressible as polynomials anymore. For example, take the non-linear operator $Q(u)(x) = u'(x)\,u(x)$. One would like to write this operator in terms of an elementary multiplication operator $M(u,v)(x) = u(x)\,v(x)$ as $Q(u) = M(D(u), u)$. Then we would have the product rule as an interaction equation $D \leftrightarrow M$, namely $D\,M(u,v) = M(D(u), v) + M(u, D(v))$. But this is not a purely operator-theoretic description anymore, since we cannot get rid of u and v. This means general rewriting is necessary now: we need substitution in addition to replacement (reduction of polyomials is replacement on equivalence classes). Maybe this could be handled by a combination of Gröbner bases and the Knuth-Bendix algorithm. Actually this is a rather subtle topic, but there are promising results recently [1, 20].

Orthogonal to these generalizations, one could also investigate *weak solutions*. In this paper, we have only considered classical solutions, but the results also make sense in a more general Sobolev setting. On the one hand, this simply changes domain and codomain of some operators; this does not harm the polynomial formulation, since it abstracts from all topological notions. On the other hand, the solution concept itself must be modified by introducing suitable testing functions v and partial integrations. Logically this means that we have a universal quantifier over v on top of the equations, so we cannot take v as an indeterminate. New ideas are necessary again.

Apart from these generalizations, there is another issue that may be worth investigating. We have already observed after Equation (12) that the concept of polynomial is not fully adequate for capturing operator composition, since it does not restrict the admissible combinations. This becomes even more apparent when we introduce operators like div and grad. In this case, we would like to distinguish vectors from scalars. For example, the composition div mathrmgrad is admissible whereas div div does not make sense. But the question of domain adequacy is not of a purely aesthetic nature: It would prevent a great deal of unnecessary S-polynomials during the search for a Gröbner basis. We would need a notion of *restricted polynomials in* $X_1, \ldots, X_n$ such that each indeterminate X_i has an associated domain, $\mathrm{dom}(\mathrm{X_i})$, and codomain, $\mathrm{cod}(\mathrm{X_i})$, where we can build up monomials $X_i\, X_j\, X_k \cdots$ only if $\mathrm{dom}(\mathrm{X_i}) = \mathrm{cod}(\mathrm{X_j})$ and $\mathrm{dom}(\mathrm{X_j}) = \mathrm{cod}(\mathrm{X_k})$, etc. Since the structure of restricted polynomials is, by its very intention, not closed under multiplication, it figures as an algebraically rather unwieldy concept. It would be interesting to develop an alternative that combines practical needs and algebraic elegance.

Acknowledgements

This work was supported by the Austrian Science Foundation FWF under the SFB grants F1302 and F1308.

References

[1] L. Bachmair, H. Ganzinger, "Buchberger's algorithm: A constraint-based completion procedure", in *First International Conference on Constraints in Computational Logics* (J.-P. Jouannaud, Ed.), Volume 845 of Lecture Notes in Computer Science, Springer, München, 1994, 285–301.

[2] M. A. Borges, Mijail Borges, "Gröbner Bases Property on Elimination Ideal in the Noncommutative Case", in [7], 323–337.

[3] B. Buchberger, "An Algorithm for Finding a Basis for the Residual Class Ring of Zero-Dimensional Polynomial Ideal" (German). PhD Thesis, University of Innsbruck, Institute for Mathematics, 1965.

[4] B. Buchberger, "An Algorithmic Criterion for the Solvability of Algebraic Systems of Equations" (German), *Æquationes Mathematicae*, **4** (1970), 374–383.

[5] B. Buchberger, "Theory Exploration Versus Theorem Proving", invited conference talk at *Calculemus '99*, Trento, Italy, (1999). Available as RISC technical report at ftp://ftp.risc.uni-linz.ac.at/pub/techreports/1999/99-46.tar.gz.

[6] B. Buchberger, "Introduction to Gröbner Bases", in [7], 3–31.

[7] B. Buchberger, Franz Winkler (Eds.), "Gröbner Bases and Applications", London Mathematical Society, Lecture Note Series 251, Cambridge University Press, 1998.

[8] E. A. Coddington, N. Levinson, "Theory of Ordinary Differential Equations", McGraw-Hill Book Company, New York, 1955.

[9] R. Courant, D. Hilbert, "Die Methoden der mathematischen Physik", Volumes 1 & 2, Springer, Berlin, 4th edition, 1993.

[10] J.H. Davenport, Y. Siret, E. Tournier, "Computer Algebra. Academic Press", London, 1988.
[11] H.W. Engl, M. Hanke, A. Neubauer, "Regularization of Inverse Problems", Kluwer, Dordrecht, 1996.
[12] H. W. Engl, M. Z. Nashed, "New Extremal Characterizations of Generalized Inverses of Linear Operators", *Journal of Mathematical Analysis and Applications*, **82** (1981), 566–586.
[13] G. Grosche, V. Ziegler, D. Ziegler, "Teubner-Taschenbuch der Mathematik", Teubner, Stuttgart, 1996.
[14] J. W. Helton, Robert L. Miller, "The system NCAlgebra", homepage and manual at
http://math.ucsd.edu/~ncalg,
http://math.ucsd.edu/~ncalg/NCBIGDOC/NCBIGDOC.html.
[15] J. William Helton, Mark Stankus, John Wavrik, "Computer Simplification of Engineering Systems Formulas", *IEEE Trans. Autom. Control*, **43/3** (1998), 302–314.
[16] J. William Helton, John Wavrik, "Rules for Computer Simplification of the Formulas in Operator Model Theory and Linear Systems", *Operator Theory: Advances and Applications*, **73** (1994), 325–354.
[17] A. M. Krall, "Applied Analysis", D. Reidel Publishing Company, Dordrecht, 1986.
[18] W. Loud, "Some Examples of Generalized Green's Functions and Generalized Green's Matrices", *SIAM Review*, **12(2)** (1970), 194–210.
[19] Y.-K. Man, "Computing Closed Form Solutions of First Order ODEs Using the Prelle-Singer Procedure", *Journ. Symb. Comp.*, **16** (1983), 423–443.
[20] C. Marché, "Normalized Rewriting: An Alternative to Rewriting Modulo a Set of Equations", *Journ. Symb. Comp.*, **11** (1996), 1–36.
[21] M. Z. Nashed (Ed.), "Generalized Inverses and Applications", Proceedings of an Advanced Seminar Sponsored by the Mathematics Research Center, Academic Press, New York, 1976.
[22] M. Z. Nashed, "Aspects of Generalized Inverses in Analysis and Regularization, in [21], 193–244.
[23] M. J. Prelle, M. F. Singer, "Elementary First Integrals of Differential Equations", *Trans. AMS*, **279/1** (1983), 215–229.
[24] D. C. Struppa, "Gröbner Bases in Partial Differential Equations", in [7], 235–245.
[25] I. Stakgold, "Green's Functions and Boundary Value Problems", John Wiley & Sons, New York, 1979.
[26] J. Wavrik, "Rewrite Rules and Simplification of Matrix Expressions", *Computer Science Journal of Moldova*, **4/2/11** (1996).