BLOCK OPERATORS AND SPECTRAL DISCRETIZATIONS

JARED L. AURENTZ* AND LLOYD N. TREFETHEN†

Abstract. Every student of numerical linear algebra is familiar with block matrices and vectors. The same ideas can be applied to the continuous analogues of operators, functions, and functionals. It is shown here how the explicit consideration of block structures at the continuous level can be a useful tool. In particular, block operator diagrams lead to templates for spectral discretization of differential and integral equation boundary-value problems in one space dimension by the rectangular differentiation, identity, and integration matrices introduced recently by Driscoll and Hale. The templates are so simple that we are able to present them as executable Matlab codes just a few lines long, developing ideas through a sequence of 12 increasingly advanced examples. The notion of the rectangular shape of a linear operator is made mathematically precise by the theory of Fredholm operators and their indices, and the block operator formulations apply to nonlinear problems too. We propose the convention of representing nonlinear blocks as shaded. At each step of a Newton iteration for a nonlinear problem, the structure is linearized and the blocks become unshaded, representing Fréchet derivative operators, square or rectangular.

Key words. block matrix, block operator, spectral collocation, Fredholm operator, index, Chebfun, ODE, integral equation

AMS subject classifications. 65M70

1. Introduction. A standard tool of computational science is block matrices. The idea begins with block 2×2 and block diagonal systems and moves on to saddle-point systems, bordered systems, and more intricate structures of many kinds. These formulations have been important for decades and are growing more so as technologies advance that blend discretization of differential and integral operators with techniques of optimization and control. Block matrices have proved useful in part because they leverage our powerful built-in visual abilities. For examples of this powerful and compelling way of thinking, see [1, 10, 22].

Traditionally, block structures are most often exploited after matrices have been obtained by discretization of differential or integral operators. In this paper we will show how they can provide a powerful framework before discretization, too, in the setting of the operators themselves, both linear and nonlinear. Applied at this stage in formulating a problem, block structures can help guide linearization and discretization. Under the leadership of Toby Driscoll, Nick Hale, and Ásgeir Birkisson, this point of view has been exploited since 2010 by Chebfun, which solves ODEs and integral equations by using the discretizations presented in this paper embedded in a system for adaptive determination of grid parameters [3, 8, 9, 14]. Ideas related to block operators have also been used in ApproxFun [16] and by other authors too, including Tretter [20], and we would certainly not claim that the Chebfun team is the only group to have made use of block structures at the operator level. Indeed, an example appears in a paper by Born, Heisenberg, and Jordan from the legendary quantum mechanical year 1926 [4], reproduced in Figure 1.1.

*jaredaurentz@gmail.com, Instituto de Ciencias Matemáticas, Nicolás Cabrera 13–15, Campus de Cantoblanco, UAM, 28049 Madrid, Spain.

†trefethen@maths.ox.ac.uk, Mathematical Institute, University of Oxford, Oxford, OX2 6GG, UK. Both authors were supported by the European Research Council under the European Union's Seventh Framework Programme (FP7/2007–2013)/ERC grant agreement no. 291068. The views expressed in this article are not those of the ERC or the European Commission, and the European Union is not liable for any use that may be made of the information contained here.

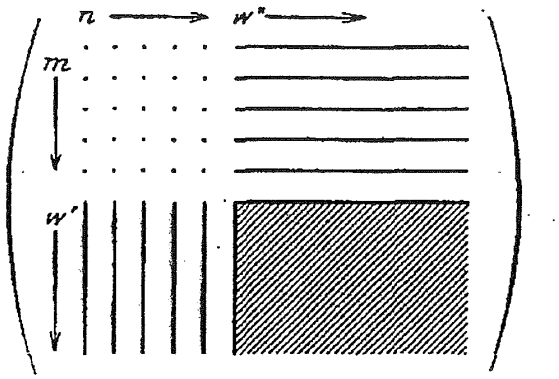

FIG. 1.1. *An early block operator diagram from Born, Heisenberg, and Jordan in* 1926 [4].

We wish to propose that explicit consideration of block operators and associated diagrams should be a central tool in scientific computing, a tool to be employed systematically in specifying discretizations, at least for 1D problems.

Our starting point is a trio of familiar discrete-continuous analogues:

$$\begin{aligned} \text{column vector} &\leftrightarrow \text{function} \\ \text{row vector} &\leftrightarrow \text{functional} \\ \text{matrix} &\leftrightarrow \text{operator} \end{aligned}$$

One naturally thinks of linear functionals and operators, but we shall propose these structures for the investigation of nonlinear functionals and operators too. A key aspect of our formulation, following Driscoll and Hale [8], is that operators are represented by blocks that are in general not square but rectangular. Although block structures can be applied more generally, these ideas have been developed in the context of a particular highly effective method of discretization of integral and differential operators with smooth (or piecewise smooth) coefficients and solutions: Chebyshev spectral collocation. In this context, block operator diagrams carry immediate implications as to the appropriate dimensions of matrix blocks constructed by discretization.

We shall present block operators through a sequence of examples, with variations introduced successively along the way including variable coefficients, eigenvalues, integral constraints, unknown parameters, periodic domains, and nonlinearity. Our convention is to represent a nonlinear object by a shaded block and a linear object by an unshaded one. This echoes the notion that a nonlinear object is a "black box" mapping inputs to outputs in an arbitrary manner. When a nonlinear problem is solved by iteration of linear problems, the linearizations involve unshading the shaded boxes.

All the problems we shall discuss involve just one space dimension—ODEs and integral equations rather than PDEs. Block operators are not limited in principle to 1D, but there is no doubt that the subject becomes more complicated in higher dimensions, just as is the case with block matrices. We regard it as an open problem to extend the framework presented here to higher dimensions.

The second author wishes to add a remark concerning the potential use of this

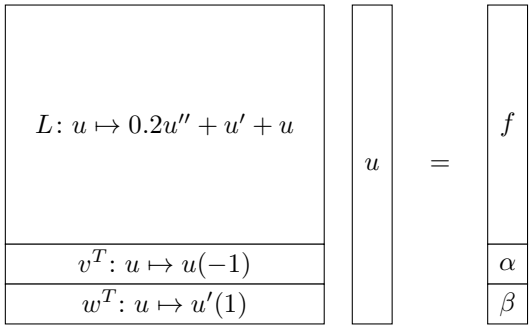

FIG. 2.1. *Block operator diagram for the linear two-point BVP* (2.1)*, Example* 1*. The rectangular shape of the block for* L *reflects its nonzero index* $\mathrm{ind}(L) = 2$ *as defined in Section* 3*.*

article in the classroom. The little book *Spectral Methods in MATLAB* is very appealing, and has attracted a wide following [18]. However, the rectangular discretizations described in this article—due to Driscoll and Hale [8]—are conceptually richer, mathematically better founded, and computationally more flexible than the square matrix methods in that book and in much of the literature of spectral collocation methods. If he were teaching this material today, after warming the students up perhaps with a few chapters of the blue book, he would turn to this article.

2. Linear constant-coefficient BVP (Example 1). As our first example, consider the advection-diffusion boundary-value problem (BVP)

$$0.2u''(x) + u'(x) + u(x) = f(x), \quad u(-1) = \alpha, \quad u'(1) = \beta, \tag{2.1}$$

consisting of a linear differential operator acting on twice continuously differentiable functions on $[-1, 1]$ together with two boundary conditions (a precise mathematical setting is specified in the next section). The function f and the numbers α and β are given. We propose to represent this problem by the diagram shown in Figure 2.1.

At the highest level, the first thing one sees in the diagram is what looks like a familiar "$Ax = b$" matrix structure. However, in fact it represents continuous objects. This is a linear equation, and the function u is the unknown. The data on the right are the function f and the two scalars α and β. The square block on the left consists of the differential operator L and the linear functionals v^T and w^T. Note that we use a linear algebra notation suggesting row vectors, though the functionals are actually continuous objects. Following what we trust is the obvious interpretation of the matrix-vector product, the diagram encodes three equations:

$$Lu = f, \quad v^T u = \alpha, \quad w^T u = \beta. \tag{2.2}$$

The two inner products represent applications of the specified linear functionals, which could be regarded as integrals involving Dirac delta functions:

$$v^T u = \int_{-1}^{1} \delta(x+1)u(x)dx, \quad w^T u = -\int_{-1}^{1} \delta'(x-1)u(x)dx. \tag{2.3}$$

A key feature of the diagram is that since the block on the left is square overall, and contains two special rows, the differential operator part of it is rectangular. This

is an observation of both mathematical and algorithmic substance. Mathematically, the rectangular shape reflects the fact that L has a nullspace of dimension 2. It is a Fredholm operator of index 2; see the next section. Speaking loosely, we say that L has dimensions "$\infty \times (\infty + 2)$" and the functionals have dimensions "$1 \times (\infty + 2)$". (There is a reason why we choose $\infty \times (\infty + 2)$ rather than $(\infty - 2) \times \infty$, concerning consistency of dimensions when there are multiple blocks, and this is explained in Example 5 in Section 7.)

Algorithmically, if we discretize this problem to form a matrix approximation, the diagram suggests that L should be discretized by a matrix with two more columns than rows. This is precisely the right thing to do in many applications, and it is the point of view at the heart of this paper. Specifically, though this is not the only reasonable choice, we shall speak of discretizations based on the method of spectral collocation, whose enabling idea is global interpolation (as opposed to, for example, local finite differencing) [8, 24]. All matrix blocks considered in this paper are defined as follows:

SPECTRAL DISCRETIZATION SCHEMA

(1) *Sample on grid;*
(2) *Apply operator:*
  (i) *for differentiation or interpolation, via interpolant,*
  (ii) *for* $+$, $-$, $*$, $/$, *or* $f(\cdot)$, *via grid samples;*
(3) *Interpolate again to get values on another grid, or at points of interest.*

This schema applies to both the operator and the functionals in Figure 2.1. Usually the domains are nonperiodic, and in this case we employ Chebyshev grids and polynomial interpolants [19]. For periodic domains, we use equispaced grids and trigonometric interpolants [23]. Algorithms and formulas for generating the Chebyshev matrix entries, which we shall not discuss, are presented in [8] and [24]. At an abstract level, since we do not give these details, our discussion is agnostic as to whether the grids employed correspond to Chebyshev points of the first kind (interior to the problem domain), Chebyshev points of the second kind (which include the endpoints), or other choices, all of which may be effective in practice. However, this paper is organized around 12 examples that are not abstract at all. Each of these is presented in what may look like pseudocode but is in fact executable code based on the Matlab commands `diffmat`, `diffrow`, `intmat`, `introw`, and `gridsample`, which are available in Chebfun.[1] These codes are based on the standard default Chebfun grids, consisting of Chebyshev points of the second kind. For the underlying approximation theory, see [19].

As an illustration of the spectral discretization schema, consider the notion of a *rectangular identity matrix,* which results from taking the identity operator in step 2 of the schema. The $n \times (n+2)$ identity, for example, is the dense "spectral downsampling" matrix corresponding to interpolation on an $(n + 2)$-point grid followed by sampling on an n-point grid. Such matrices have been used by Driscoll and Hale [8] and appear in most of our examples. Here for example is the 4×6 identity on $[-1, 1]$, which we obtain by calling `diffmat` with an argument 0, corresponding to differentiation 0 times:

[1] As of August 2016, these commands are available in the development branch of Chebfun at GitHub. To keep the article from getting too long, code listings for examples 5, 6, and 12 have been omitted.

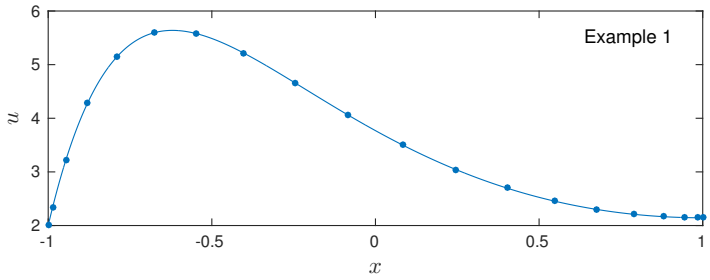

FIG. 2.2. *Solution of* (2.1) *by the* $(n+2)\times(n+2)$ *block matrix system of equations* `A*u = rhs` *derived by spectral discretization of Figure* 2.1 *with* $n = 18$*, which is enough for accuracy close to machine precision. The dots show the entries of the solution vector* `u`*, with abscissas at Chebyshev points, and the curve is the corresponding polynomial interpolant.*

```
>> diffmat([4,6],0)
ans =
    1.0000         0         0         0         0         0
   -0.1500    0.4854    0.7854   -0.1854    0.1146   -0.0500
   -0.0500    0.1146   -0.1854    0.7854    0.4854   -0.1500
         0         0         0         0         0    1.0000
```

From the block diagram of Figure 2.1, via the spectral discretization schema, one comes to a prescription for discretizing (2.1). Given $n \geq 1$, we generate an $(n+2)\times(n+2)$ discretization matrix A like this:

```
% Example 1
L = 0.2*diffmat([n n+2],2) + diffmat([n n+2],1) + diffmat([n n+2],0)
vT = diffrow(n+2,0,-1), wT = diffrow(n+2,1,1)
A = [L; vT; wT]
```

The three calls to `diffmat` produce $n\times(n+2)$ matrices whose effects are equivalent to (1) interpolating the data by a degree $n+1$ polynomial on an $(n+2)$-point grid; (2) differentiation 2, 1, or 0 times, respectively; and (3) sampling the result on an n-point grid. The row vectors `vT` and `wT` have the effect of (1) interpolation as before, (2) differentiation 0 or 1 times, respectively, and (3) evaluation of the result at the point -1 or 1, respectively. If `f` is a function that computes f, the right-hand side vector for the discrete approximation of Figure 2.1 can be generated by

```
rhs = [gridsample(f,n); alpha; beta] ,
```

where `gridsample(f,n)` samples the function f on the grid of n points, and we compute the discrete spectral collocation solution by executing `u = A\rhs`.

The solution n-vector `u` corresponding to the choices $n = 18$, $f(x) = e^x$, $\alpha = 2$, $\beta = 0$ is plotted in Figure 2.2. This value of n is enough to resolve the problem to close to full 16-digit machine precision as measured by the norm of the residual `A*u-rhs`, or at the continuous level, by the difference between the two sides of Figure 2.1. The matrix A has a condition number of about 4.52×10^4, and the computed vector u and its polynomial interpolant are accurate to about 13 digits. All our computed examples will follow this pattern of choosing a value of n that is roughly the smallest that brings

the residual down to approximately machine epsilon in 16-digit arithmetic.

The solution vector we have just computed corresponds to the black dots in Figure 2.2. The curve between the dots is the polynomial interpolant (here of degree $n+1$ since the number of dots is $n+2$). The standard stable algorithm for computing this interpolant is the barycentric interpolation formula [2, 19]. We invoke this formula via the Chebfun command `chebfun(u)`, which converts a vector of values at Chebyshev points to the interpolating polynomial. Thus a complete executable Chebfun code to produce Figure 2.2 (omitting formatting commands) looks like this:

```
% Example 1
f = @(x) exp(x); alpha = 2; beta = 0; n = 18;
L = 0.2*diffmat([n n+2],2) + diffmat([n n+2],1) + diffmat([n n+2],0);
vT = diffrow(n+2,0,-1); wT = diffrow(n+2,1,1);
A = [L; vT; wT];
rhs = [gridsample(f,n); alpha; beta];
u = A\rhs;
plot(chebfun(u),'.-')
```

We shall not give complete codes like this for the remaining examples, but they can all be found in the online supplementary materials. It takes about 5 seconds total on a 2015 desktop machine to run all of the codes.

3. Rectangular blocks and the index of an operator. Most of this paper is concrete and tutorial in style, developing our subject with a sequence of computed examples. In this section, we briefly present some mathematical foundations.

The notion of the rectangular shape of a block can be made precise via the theory of Fredholm operators [15, 17, 25]. The linear operator L of (2.1), for example, can be regarded as a closed operator in the Banach space $X = C([-1,1])$ of continuous functions in the maximum norm. The domain of L, $D(L)$, is the subset of X consisting of twice continuously differentiable functions on $[-1,1]$, and the range, $R(L)$, is all of X. The nullspace is the kernel of L in $D(L)$, namely the span of the functions $e^{a_\pm x}$ with $a_\pm = (-5 \pm \sqrt{5})/2$, and the *nullity* of L, $\mathrm{nul}(L)$, defined as the dimension of this nullspace, is equal to 2. The *deficiency* of L, $\mathrm{def}(L)$, is the codimension of the range $R(L)$ in X, which in this case is 0 since $L(D(L)) = X$. The *index* of L is the difference

$$\mathrm{ind}(L) = \mathrm{nul}(L) - \mathrm{def}(L), \tag{3.1}$$

assuming at least one of these quantities is finite. (In general a linear operator L is a *Fredholm operator* if $R(L)$ is closed and both $\mathrm{nul}(L)$ and $\mathrm{def}(L)$ are finite.) Thus for (2.1) we have $\mathrm{ind}(L) = 2 - 0 = 2$, giving a mathematically precise measure of the degree of rectangularity of the block depicting L. When discretized, L will be approximated by matrices of dimension $n \times (n+2)$ for $n \geq 1$.

If the rectangular operator block L has index 2, what about the full square object $\mathcal{L}$ with its two extra rows? As one would expect, this can be regarded as a linear operator of index zero, which accordingly defines a problem with a unique solution. The domain space $C([-1,1])$ and domain $D(L)$ are the same as before, and the range is now $C([-1,1]) \oplus \mathbf{R}^2$ since application of $\mathcal{L}$ produces two scalars as well as a function. If we define the norm in $C([-1,1]) \oplus \mathbf{R}^2$ as the maximum of the norms of its components in $C([-1,1])$, $\mathbf{R}$, and $\mathbf{R}$, then $\mathcal{L}$ is a closed operator with $\mathrm{nul}(\mathcal{L}) = \mathrm{def}(\mathcal{L}) = \mathrm{ind}(\mathcal{L}) = 0$.

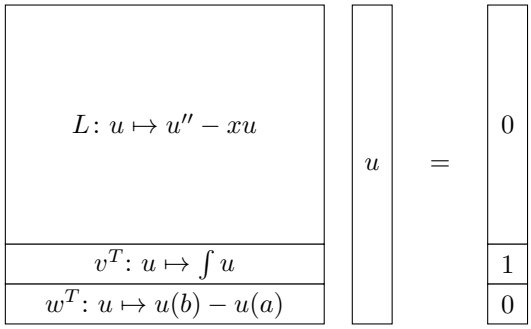

FIG. 4.1. *Variable coefficient BVP* (4.1)*, Example* 2.

Throughout this paper, boundary conditions are imposed on problems involving operators, but they are not part of the definitions of the operators. For example, a second-order differential operator L may have its domain $D(L)$ restricted to twice-continuously differentiable functions on $[-1,1]$, but $D(L)$ will not be restricted further by any boundary conditions such as $u(-1) = 0$. In other words, boundary conditions are constraints on a solution, not on a function space. This approach fits naturally with the block operator formulations, and in our experience it avoids confusion in problems involving adjoints. A problem involving an operator of a certain index and some boundary conditions will have an adjoint problem involving the (formal) adjoint operator, of the same index, and a set of adjoint boundary conditions. See examples 11 and 12.

For a nonlinear operator $N\colon u \mapsto N(u)$, as we shall begin to consider with Example 9, a basic tool of analysis and of algorithms is the Frećhet derivative $N' = \partial N/\partial u$. If N is continuously Fréchet-differentiable as a function of u and $\mathrm{ind}(N'(u))$ is independent of u, then this number is called the index of N [25, Sec. 8.4]. This condition of constancy of $\mathrm{ind}(N'(u))$ holds for many nonlinear differential operators that arise in practice, where the leading order term is linear with nonvanishing coefficient; our examples 9, 10, and 12 are in this category. We use the index terminology throughout this article, though our focus is not on a rigorous theory.

4. Variable coefficients and nonstandard BCs (Example 2). Our second example is another linear BVP, an Airy equation:

$$u''(x) - xu(x) = 0, \quad \int_a^b u(s)ds = 1, \quad u(a) = u(b). \tag{4.1}$$

This problem differs from (2.1) in having a variable coefficient and also in having two less standard "boundary" conditions, one imposing an integral condition, the other coupling the left and right boundaries. It is also set on a different domain, $[a,b]$ instead of $[-1,1]$.

The block operator diagram for (4.1), shown in Figure 4.1, has the same structure as in Figure 2.1 but with different entries in the blocks. First, the domain is now $[a,b]$ instead of $[-1,1]$ and L is now the map $L\colon u \mapsto u'' - xu$, again with index 2. Second, the functionals v^T and w^T now evaluate an integral and a difference of endpoint

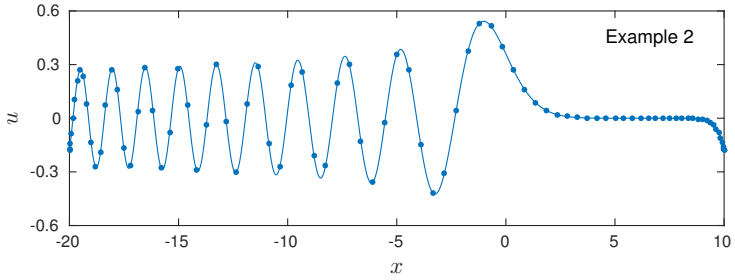

FIG. 4.2. *Solution of* (4.1) *by the* $(n+2)\times(n+2)$ *block matrix system of equations derived from Figure* 4.1 *with* $n = 85$.

values, respectively:

$$v^T u = \int_a^b u(x)dx, \quad w^T u = \int_a^b [\delta(x-b) - \delta(x-a)]u(x)dx. \tag{4.2}$$

As always, the point is that the schematic figure should provide a template for discretization. This can be constructed as follows, with the new variable `X` now appearing as an additional argument in each call to `diffmat` to specify a domain that differs from the default $[-1,1]$. (This `X` has nothing to do with the Banach space X of the last section.)

```
% Example 2
X = [a,b]; x = @(x) x
L = diffmat([n n+2],2,X) - diag(gridsample(x,n,X))*diffmat([n n+2],0,X)
vT = introw(n+2,X), wT = diffrow(n+2,0,b,X) - diffrow(n+2,0,a,X)
A = [L; vT; wT], rhs = [zeros(n,1); 1; 0], u = A\rhs
```

The call to `diffrow` with order -1 in the definition of `vT` produces a row $(n+2)$-vector corresponding to the discretization of the integral by the spectral discretization schema. (In other words, it returns a row vector of quadrature weights, which for the case of Chebyshev grids will be Clenshaw–Curtis weights; see chapter 19 of [19].) The function `gridsample` in the construction of `L` samples the function x on the n-point grid, and the left-multiplication by the square diagonal matrix with these entries produces the linear algebra required by the spectral discretization schema. (As a practical matter there is no need to carry out a dense matrix multiplication, and this can be avoided in Matlab by replacing `diag(...)` by `spdiags(...,0,n,n)`, which invokes the more efficient sparse matrix representation of the diagonal matrix.) Figure 4.2 shows the result for $a = -20$, $b = 10$ computed with $n = 85$, about what is needed for full resolution in 16-digit arithmetic. The matrix A has a condition number of about 1.03×10^5, and the solution is accurate to 12 digits. We remind the reader that executable codes for this and subsequent examples can be found in the supplementary materials.

5. Eigenvalue problem (Example 3). Linear algebraists speak of eigenvalue problems $Ax = \lambda x$ and generalized eigenvalue problems $Ax = \lambda Bx$ (and associated pencils $A - \lambda B$) [10, 22]. As explained in [8], the generalized setting is usually the appropriate one for formulating operator eigenvalue problems, even nongeneralized ones, by block operators.

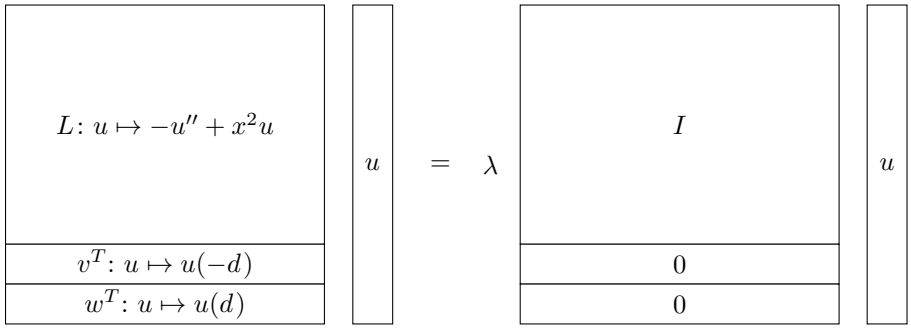

FIG. 5.1. *Harmonic oscillator eigenvalue problem* (5.1)*, Example* 3*. The operator* I *on the right is a rectangular identity, which is discretized by an* $n \times (n+2)$ *spectral downsampling matrix.*

For example, consider the Schrödinger harmonic oscillator eigenvalue problem

$$-u''(x) + x^2 u(x) = \lambda u, \quad u(-d) = u(d) = 0. \tag{5.1}$$

A block operator diagram for this problem appears in Figure 5.1. Note that it includes a rectangular identity operator on the right-hand side, necessary so that both sides can act in the same fashion on the vector u so that the diagram has eigenvalue structure.

The diagram immediately tells us the matrix discretization as a generalized eigenvalue problem—in Matlab notation, `eig(A,B)` for square matrices A and B. Note that `diffmat` is called with an argument of 0 to produce a rectangular identity matrix of the kind discussed in §2.

```
% Example 3
X = [-d,d], x2 = @(x) x.^2
L = -diffmat([n n+2],2,X) + diag(gridsample(x2,n,X))*diffmat([n n+2],0,X)
vT = diffrow(n+2,0,-d,X), wT = diffrow(n+2,0,d,X)
A = [L; vT; wT]
I = diffmat([n n+2],0,X)
B = [I; zeros(2,n+2)]
```

With $d = 10$ and $n = 80$, the first six eigenvalues of this matrix pencil match those of (5.1) with $d = \infty$, namely $1, 3, \ldots, 11$, to about 14 digits. Figure 5.2 shows the corresponding eigenvectors plotted in a fashion familiar to physicists, each one raised by a distance equal to the corresponding eigenvalue.

It is interesting to consider Figure 5.1 from the point of view of Fredholm operators. The block representing L is a Fredholm operator with index 2, but the corresponding block on the right representing I, though also rectangular, would have index 0 by itself. It is the combination $u \mapsto -u'' + x^2 u - \lambda u$ that has index 2 (for any λ).

6. Periodic problems (Example 4). Sometimes an ODE is posed on a periodic domain. One could regard such a situation as a nonperiodic problem with boundary conditions to enforce periodicity, such as $u(a) = u(b)$ and $u'(a) = u'(b)$ for a second-order equation on $[a, b]$. It is more natural, however, to regard the domain as intrinsically periodic, with no boundary and no boundary conditions. For example,

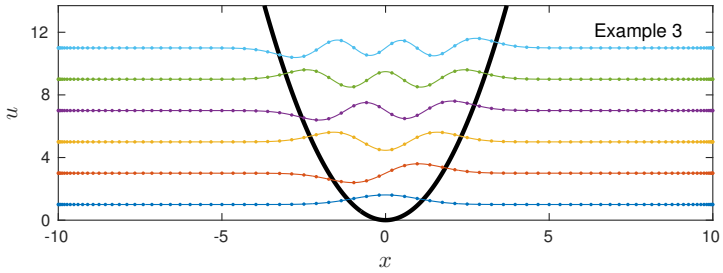

FIG. 5.2. *First six eigenfunctions of* (5.1) *on* $[-d, d]$ *with* $d = 10$ *computed by the* $(n + 2) \times (n + 2)$ *block matrix eigenvalue problem derived from Figure* 5.1 *with* $n = 80$.

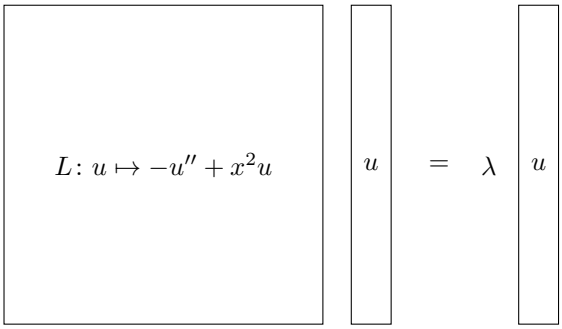

FIG. 6.1. *Harmonic oscillator eigenvalue problem* (6.1)*, now posed on the periodic interval* $[-d, d]$*, Example* 4*. There are no boundary conditions and no special rows, and* $\mathrm{ind}(L) = 0$.

suppose we pose the harmonic oscillator problem (5.1) again, but now on a periodic domain $[-d, d]$:

$$-u''(x) + x^2 u(x) = \lambda u, \quad -d \le x \le d, \quad \text{periodic.} \tag{6.1}$$

(The coefficient function x^2 is continuous across the point of periodicity, $x = \pm d$, though not continuously differentiable.) The block operator diagram, shown in Figure 6.1, is trivial, with the operator represented by a square block for a change. This is consistent with the Fredholm theory, since the operator $L: u \mapsto -u'' + x^2 u$ has the trivial nullspace on the space of twice continuously differentiable periodic functions on $[-d, d]$ in the maximum norm, making the index zero.

To solve a periodic problem, one follows the spectral discretization schema with an equispaced grid and a trigonometric interpolant: a Fourier spectral method [5, 23].

```
% Example 4
X = [-d,d], x2 = @(x) x.^2
L = -diffmat([n n],2,X,'trig') + diag(gridsample(x2,n,X,'trig'))
eig(L)
```

There is a practical reason for considering the periodic oscillator (6.1), namely that its lower eigenfunctions will be virtually identical to those of (5.1), thanks to exponential decay for large $|x|$, and somewhat easier to compute. Figure 6.2 shows the first six eigenfunctions computed in this manner, matching Figure 5.2. Since the

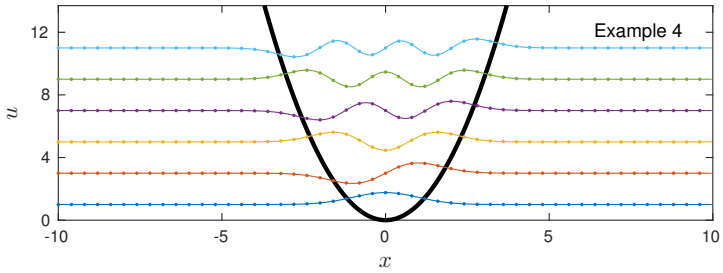

FIG. 6.2. *Repetition of Figure* 5.2*, but now for a solution on an equispaced grid for the periodic formulation of* (6.1) *and Figure* 6.1 *with* $n = 50$*. Note that the equally spaced grid has about the same spacing in the middle as the Chebyshev grid of Figure* 5.1*.*

spacing of an n-point periodic grid is about $\pi/2$ times denser in the middle than that of an n-point Chebyshev grid, $n = 50$ points now suffice for 14-digit precision.

Of course there are also ODE problems with genuine periodicity, as opposed to this example of the use of a periodic domain for computational convenience. Example 10 below is of this kind.

7. Coupled system of BVPs (Example 5). If a BVP involves more than one variable, there will be more than one block in its block operator diagram. For example, the second-order scalar wave equation

$$u'' = -300e^{3x}u, \quad u(-1) = 0, \quad u(1) = 1$$

is equivalent to the pair of first-order equations

$$u' = 10v, \quad v' = -30e^{3x}u, \quad u(-1) = 0, \quad u(1) = 1 \tag{7.1}$$

derived by introducing the variable $v = u'/10$. A block operator diagram for this system is shown in Figure 7.1, containing two rectangular differentiation operators, a rectangular identity (times -10), and a rectangular operator of multiplication by $30e^{3x}$. (Careful readers will note that we are not completely consistent in the notation we use in our diagrams, making different choices for different examples to communicate most economically.) The solution computed with $n = 90$ is shown in Figure 7.2. (There is no reason why u and v have to have the same values of n, incidentally.)

In Figure 7.1 as in Figure 6.1, rectangular blocks appear that do not in themselves represent operators of positive index, namely the identity and the multiplication operator. The rectangularity of each block is determined by the order of the highest derivative in its column. If the map $v \mapsto v'$ in Figure 7.1 were $v \mapsto v''$, for example, then that block and the -10 block above it would have dimensions "$\infty \times (\infty+2)$", but the two blocks on the left applied to the variable u would still be of size "$\infty \times (\infty+1)$". This rule of sizing of blocks was introduced by Driscoll and Hale [8].

8. Integral operators (Examples 6 and 7). With integral as opposed to differential operators, one has the possibility of nontrivial deficiency (range codimension), giving a negative index. For example, consider the problem

$$\int_{-1}^{x} u(t)dt + c = f(x) \tag{8.1}$$

$u \mapsto u'$	-10	u		0
$30e^{3x}$	$v \mapsto v'$		=	0
$u \mapsto u(-1)$		v		0
$u \mapsto u(1)$				1

FIG. 7.1. *Block operator diagram for the coupled system* (7.1)*, Example* 5*. Blank rows represent zero functionals. Since first-order differentiation has index* 1*, the four blocks are each of dimensions "*$\infty \times (\infty + 1)$*".*

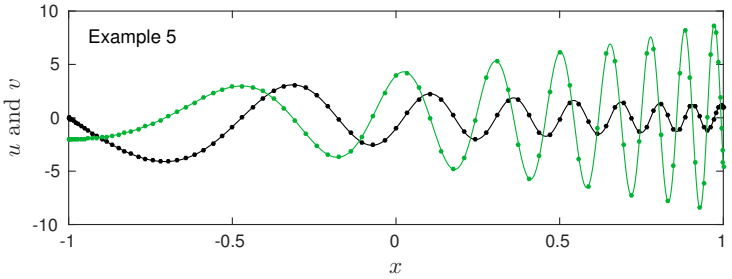

FIG. 7.2. *Solution of* (7.1) *by the* $2n \times 2n$ *block matrix system of equations derived from Figure* 7.1 *with* $n = 90$*.*

where f is given and the function u and constant c are unknown. If we take the underlying Banach space X to be $C([-1, 1])$, then the indefinite integration operator $L\colon u \mapsto \int^x u$ has $\mathrm{def}(L) = 1$ since its range is limited to functions that take the value 0 at the left endpoint. Thus the index is $\mathrm{ind}(L) = -1$, and in a block operator formulation, as shown in Figure 8.1, this part of the diagram will be a rectangle taller than it is wide. A discretization will involve an $n \times (n-1)$ spectral integration matrix. (In practice one would not normally solve this problem with a matrix since u can be obtained by simply differentiating f, so we do not show a code or a figure.)

For an example with more substance, following Clenshaw, Greengard, and Driscoll [6, 7, 11], let us formulate an integral equation BVP equivalent to the advection-diffusion equation (2.1) by rewriting that problem in terms of the variable $v = u''$. This change of variables implies

$$u'(x) = \int_{-1}^{x} v(t)dt + c, \quad u(x) = \int_{-1}^{x}\int_{-1}^{t} v(s)ds\,dt + (x+1)c + d \tag{8.2}$$

for some integration constants c and d, so (2.1) becomes

$$0.2v(x) + \int_{-1}^{x} v(t)dt + c + \int_{-1}^{x}\int_{-1}^{t} v(s)ds\,dt + (x+1)c + d = f(x), \tag{8.3}$$

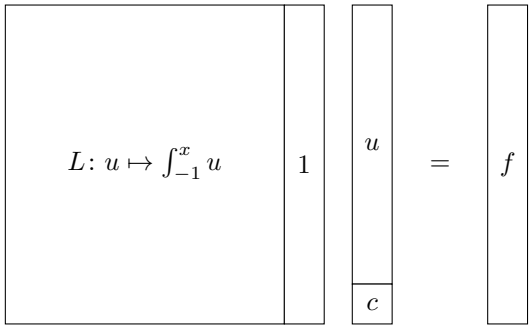

FIG. 8.1. *Block operator diagram for the indefinite integral problem* (8.1)*, Example* 6*. Here the index is* $\mathrm{ind}(L) = -1$.

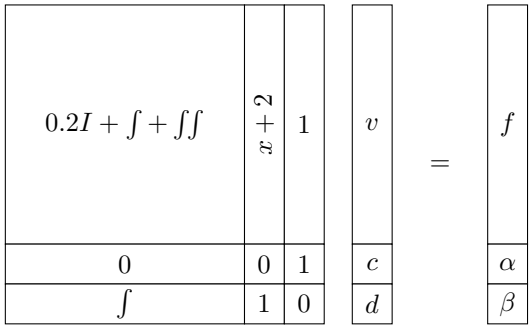

FIG. 8.2. *Block operator diagram for the integral operator problem* (8.3)–(8.4) *that arises when* (2.1) *is rewritten in terms of the variable* $v = u''$*, Example* 7*. The symbols* $\int$ *and* $\iint$ *in the operator block represent the operators of single and double indefinite integration from the left endpoint, respectively, and the symbol* $\int$ *in the functional row represents definite integration.*

together with the boundary conditions

$$d = \alpha, \quad \int_{-1}^{1} v(t)dt + c = \beta. \tag{8.4}$$

A block operator diagram for this problem is shown in Figure 8.2, containing the second appearance in this paper of a square block; see the similar formulation in [7]. The computed solution shown in Figure 8.3 is based on this discretization:

```
% Example 7
L = 0.2*intmat(n,0) + intmat(n,1) + intmat(n,2)
vT = zeros(1,n), wT = introw(n)
A = [ [L; vT; wT] [gridsample(x+2,n); 0; 1] [ones(n,1); 1; 0] ]
rhs = [gridsample(f,n); alpha; beta]
vcd = A\rhs, v = vcd(1:n), c = vcd(n+1), d = vcd(n+2)
```

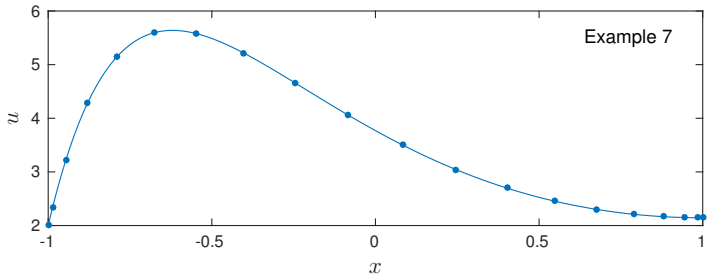

FIG. 8.3. *Solution of* (8.3)–(8.4) *by the* $n \times n$ *block matrix system of equations derived by spectral discretization of Figure* 8.2 *with* $n = 18$. *For comparison with Figure* 2.2, *the plot shows not* v *but its second integral* u *as defined by* (8.2).

9. Optimization with a scalar constraint (Example 8). Consider the constrained optimization problem

$$\min_u \frac{1}{2}\int_{-1}^{1} w(x)(u(x)-f(x))^2 dx, \quad \int_{-1}^{1} g(x)u(x)dx = c, \tag{9.1}$$

where the scalar c and the functions $f, w, g \in C([-1,1])$ are given, with $w(x) > 0$ for all x. If f^T and g^T are the bounded linear functionals on $C([-1,1])$ defined by

$$f^T\colon u \mapsto \int_{-1}^{1} f(x)u(x)dx, \quad g^T\colon u \mapsto \int_{-1}^{1} g(x)u(x)dx, \tag{9.2}$$

and the functional u^T is defined analogously for any fixed $u \in C([-1,1])$, then the problem is equivalent to

$$\min_u E(u), \quad E(u) = \frac{1}{2}u^T W u - f^T W u, \quad g^T u = c, \tag{9.3}$$

where W is the operator of multiplication by w. A standard way to solve such a problem is to introduce the Lagrangian function

$$\mathcal{L}(u,\lambda) = \frac{1}{2}u^T W u - f^T W u + (g^T u - c)\lambda, \tag{9.4}$$

where λ is a scalar Lagrange multiplier. The partial derivative of $\mathcal{L}$ with respect to λ and its gradient with respect to u are

$$\frac{\partial \mathcal{L}}{\partial \lambda} = g^T u - c, \quad \nabla_u \mathcal{L} = Wu - Wf + \lambda g. \tag{9.5}$$

Suppose we find a stationary point of $\mathcal{L}(u,\lambda)$ with respect to both λ and u. Then $\partial\mathcal{L}/\partial\lambda = 0$, so the constraint $g^T u = c$ is satisfied, and $\nabla_u \mathcal{L} = 0$, which implies $Wu - Wf = -\lambda g$, that is, $\nabla_u E(u) = -\lambda g$. Together these are the *Karush–Kuhn–Tucker (KKT) conditions.* Thus the gradient of E points in the direction of g, i.e., orthogonally to the constraint hyperplane defined by the equation $g^T u = c$. So at the point u, $E(u)$ restricted to the hyperplane has a stationary point. This is the reasoning behind the standard method of setting the equations (9.5) to zero to solve a problem of this kind. Figure 9.1 gives the block diagram, which has the form known as a saddle-point system [1].

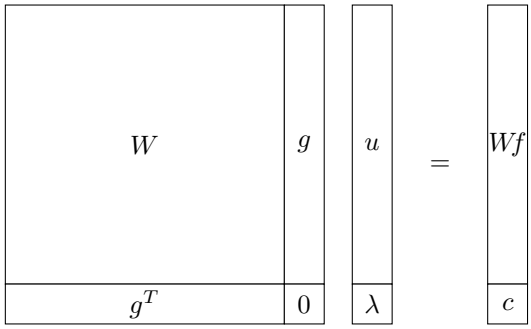

FIG. 9.1. *Block operator diagram to set the two expressions of* (9.5) *to zero to find a stationary point of the Lagrangian, a standard method of solving* (9.1)*, Example* 8.

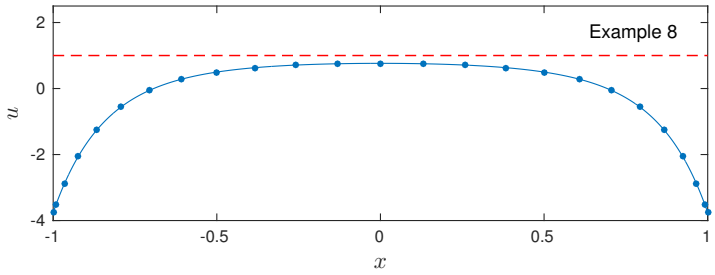

FIG. 9.2. *Solution of* (9.1) *with* $f(x) = g(x) = 1$, $w(x) = \exp(-3x^2)$, *and* $c = 0$ *by the* $n \times n$ *block matrix system of equations* `A*ulam = rhs` *derived by spectral discretization of Figure* 9.1 *with* $n = 25$. *The dashed line shows* f. *The solution is the closest function to* f, *measured by the least-squares norm with weight* w, *whose integral is zero.*

A code segment for discretization of Figure 9.1 with $f(x) = g(x) = 1$, $w(x) = \exp(-3x^2)$, and $c = 0$ is given below, and the result is plotted in Figure 9.2. The Lagrange multiplier takes the value $\lambda \approx 0.2368426790707$.

```
% Example 8
gT = introw(n)
A = [diag(gridsample(w,n)) gridsample(g,n); gT 0]
rhs = [gridsample(w,n); c]
ulam = A\rhs, u = ulam(1:n), lambda = ulam(n+1)
```

10. Nonlinear operators (Example 9). Like a linear operator, a nonlinear operator has "rectangular shape" in the sense that it maps inputs in one space to outputs in another. Within the rectangle, however, the couplings of inputs to outputs may be arbitrary. Accordingly, we propose the convention of representing a nonlinear block of an operator by a shaded rectangle. When the operator is linearized about a particular input function for a Newton iteration, we are back in the linear situation and the shading is removed, giving a block operator of the kind discussed up to now in this paper, which as usual can serve as a template for discretization.

For example, a block operator diagram for the nonlinear BVP

$$N(u) = u''(x) + u^2(x) - f(x) = 0, \quad u(-1) = \alpha, \quad u(1) = \beta \tag{10.1}$$

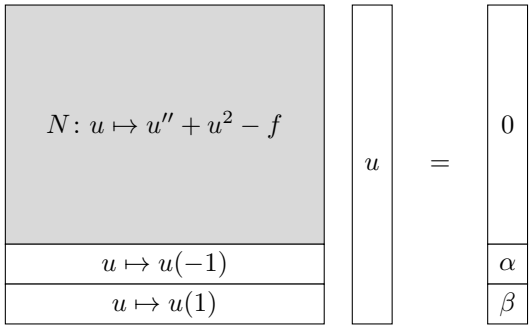

FIG. 10.1. *Block operator diagram for the nonlinear BVP* (10.1)*, Example* 9*. The nonlinear block, with* $\mathrm{ind}(N) = 2$*, is shaded as a reminder of the arbitrary nature of possible nonlinear couplings between input and output functions.*

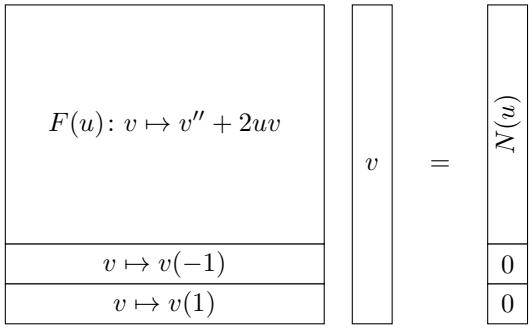

FIG. 10.2. *Linearization of* (10.1) *for a Newton iteration converts Figure* 10.1 *to a linear structure representing the update equation* (10.2)*.*

is shown in Figure 10.1. To solve this problem, one could carry out a Newton iteration of the form $u^{(k+1)} = u^{(k)} - v$ for $k \geq 0$ (see [3]), where v is the solution of the linear update equation

$$F(u^{(k)})v = N(u^{(k)}), \quad v(-1) = 0, \quad v(1) = 0. \tag{10.2}$$

(For simplicity of presentation, we assume $u^{(0)}$ satisfies the boundary conditions.) Here $F(u) = (\partial N/\partial u)(u)$ is the Fréchet derivative operator for N at u,

$$F(u)\colon v \mapsto v'' + 2uv, \tag{10.3}$$

a linear operator in $C([-1,1])$ of index 2. Equation (10.2) can accordingly be represented by an unshaded block operator diagram, Figure 10.2. Since the leading order term of $F(u)$ is constant, $\mathrm{ind}(F(u))$ is independent of u, so the nonlinear operator has a well-defined index $\mathrm{ind}(N) = 2$ according to the definition of Section 3.

The essential lines of a code segment for realizing the algorithm just described are below, and the result is shown in Figure 10.3. In Chebfun, Fréchet derivative operators are constructed automatically by a process of automatic differentiation developed by Birkisson and Driscoll [3].

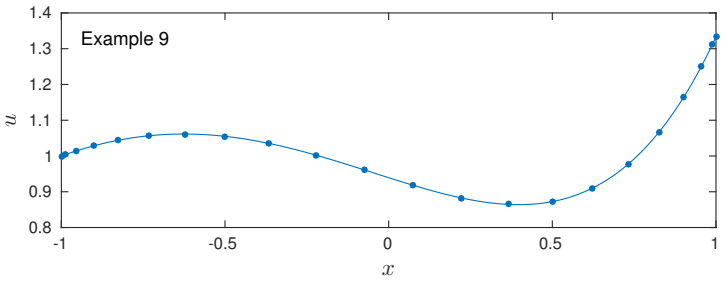

FIG. 10.3. *Solution of* (10.1) *with* $f(x) = e^{2x}$, $\alpha = 1$, $\beta = 4/3$ *based on Newton iteration* (10.2)*, with the linear problem of Figure* 10.2 *solved by spectral discretization. The iteration converges to approximately full precision (as measured by residual norm) in* 7 *steps from the initial guess* $u^{(0)}(x) = \alpha(1-x)/2 + \beta(1+x)/2$.

```
% Example 9
D2 = diffmat([n n+2],2), I = diffmat([n n+2],0)
N = @(u) D2*u + I*u.^2 - f
F = @(u) D2 + 2*I*diag(u)
A = @(u) [F(u); diffrow(n+2,0,-1); diffrow(n+2,0,1)]
rhs = @(u) [N(u); 0; 0]
for k = 1:7
   u = u - A(u)\rhs(u)
end
```

11. Nonlinear oscillator with unknown period (Example 10). Sometimes a nonlinear equation may have a periodic solution of unknown period. For example, consider the van der Pol equation

$$y'' + \mu(y^2 - 1)y' + y = 0, \quad y(0) = 1, \quad 0 \le \tau \le T, \quad \text{periodic}, \tag{11.1}$$

where $\mu > 0$ is a damping parameter, τ is the independent variable (time), and the period T is unknown. The condition $y(0) = 1$ is included to eliminate translation-invariance in t; this is known as an *anchor condition.* With this condition there will be two solutions with the same minimal value of T, one with $y'(0) > 0$ and the other with $y'(0) < 0$. In addition there will be two solutions with twice that value of T, two solutions with three times that value of T, and so on.

This problem can be recast onto the fixed interval $[0, 1]$ by defining $t = \tau/T$ and $u(t) = y(\tau)$. After multiplying by T^2 we obtain

$$N(u, T) = u'' + T\mu(u^2 - 1)u' + T^2 u = 0, \quad u(0) = 1, \quad 0 \le t \le 1, \quad \text{periodic}. \tag{11.2}$$

This is a problem where both the function u and the scalar T enter nonlinearly, and a block diagram is given in Figure 11.1.

As in the last section, we can solve the problem by the Newton iteration

$$u^{(k+1)} = u^{(k)} - v, \quad T^{(k+1)} = T^{(k)} - s \tag{11.3}$$

for $k \ge 0$, where v and s satisfy the update equations

$$\frac{\partial N}{\partial u} v + \frac{\partial N}{\partial T} s = N(u^{(k)}, T^{(k)}), \quad v(0) = 0, \tag{11.4}$$

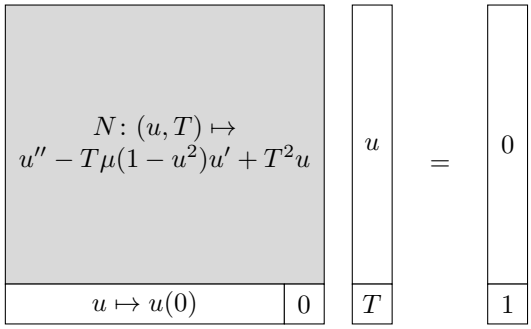

FIG. 11.1. *Block operator diagram for the van der Pol oscillator problem* (11.2) *with unknown period* T*, after transplantation to* $t \in [0,1]$*, Example* 10*. The index is* $\mathrm{ind}(N) = 1$.

shown in Figure 11.2. (We assume $u^{(0)}$ satisfies the anchor condition $u(0) = 1$.) Note that the linearization breaks the rectangular block of Figure 11.1 into a square block of index 0 together with one further column, corresponding to the linear operators

$$\frac{\partial N}{\partial u}(u,T)\colon\ v \mapsto v'' + T\mu(u^2-1)v' + (2T\mu uu')v + T^2v \tag{11.5}$$

and

$$\frac{\partial N}{\partial T}(u,T)\colon\ s \mapsto \mu(u^2-1)u's + (2Tu)s. \tag{11.6}$$

The square block is a result of the use of a periodic formulation, as in Section 6. With a nonperiodic formulation, periodicity would be imposed by boundary conditions $u(0) = u(1)$, $u'(0) = u'(1)$, and this block would be rectangular of index 2.

A spectral discretization follows the pattern of earlier examples of this article, with the essential lines of code as below. With $\mu = 2$ and the initial guess $u^{(0)}(t) = 2\cos(2\pi(t + \frac{1}{6}))$, $T^{(0)} = 5$, the Newton iteration converges in 8 steps to the solution shown in Figure 11.3, with $T = 7.6298744797$.

```
% Example 10
X = [0,1], I = eye(n)
D = diffmat([n n],1,X,'trig'), D2 = diffmat([n n],2,X,'trig')
N = @(u,T) D2*u + T*mu*(u.^2-1).*(D*u) + T^2*u
dNdu = @(u,T) D2 + T*mu*diag(u.^2-1)*D + 2*T*mu*diag(u.*(D*u)) + T^2*I
dNdT = @(u,T) mu*(u.^2-1).*(D*u) + 2*T*u
A = @(u,T) [dNdu(u,T) dNdT(u,T); diffrow(n,0,0,X,'trig') 0]
rhs = @(u,T) [N(u,T); 0]
for k = 1:8
  vs = A(u,T)\rhs(u,T), v = vs(1:n), s = vs(n+1)
  u = u - v, T = T - s
end
```

12. SVD of a differential operator (Example 11). Operator theory sprang from the investigations of integral operators by Fredholm, Hilbert, and Schmidt, and for our next example, we turn to this circle of ideas. Given $f \in C([-1,1])$ and $\varepsilon > 0$, the nonself-adjoint problem defined by

$$\varepsilon v'' + v' = f, \quad v(-1) = v(1) = 0 \tag{12.1}$$

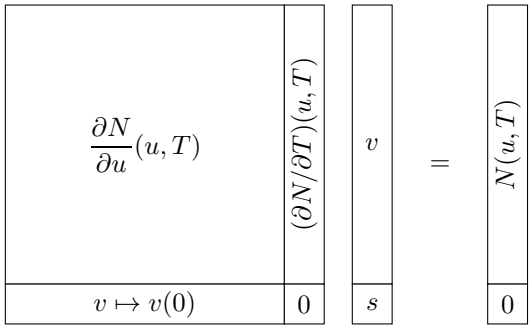

FIG. 11.2. *Linearization of Figure* 11.1 *for each Newton step. The rectangular block is now unshaded and breaks into a square block differential operator and an additional column function. Together, these pieces represent a linear operator of index* 1.

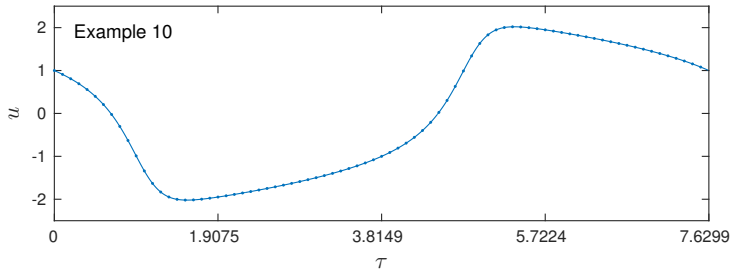

FIG. 11.3. *Solution of* (11.2) *with* $\mu = 2$ *after* 8 *Newton steps with* $n = 80$*, rescaled from the computational interval* $t \in [0,1]$ *to the problem interval* $\tau \in [0,T]$*. A different initial guess could have led to a periodic translation of this solution with* $y'(0) > 0$*, or to multi-period solutions on intervals of lengths an integer multiple of this one.*

has a unique solution v that could be represented in terms of a Fredholm integral operator, which would be compact and accordingly have singular values decaying to 0. Here, let us construct an SVD of the differential operator directly by considering coupled eigenvalue equations for the unknown functions v and u:

$$\varepsilon v'' + v' = \sigma u, \quad \varepsilon u'' - u' = \sigma v, \quad v(\pm 1) = u(\pm 1) = 0. \tag{12.2}$$

(Each singular value will appear twice among the solutions of (12.2), once with a positive sign and once negated, corresponding to a negation of the sign of v relative to u.) Note that the equation and boundary conditions for u are the adjoint of those for v, reflecting the familiar `[0 A; A' 0]` structure sometimes used to compute the SVD of a matrix `A`. A block operator diagram is shown in Figure 12.1. As with the harmonic oscillator of Figure 5.1, we have a generalized eigenvalue problem containing rectangular identity operators. The essential lines of discretization are given below, and the first three singular functions u and v are shown in Figure 12.2. The corresponding singular values are 2.4975282097, 4.2132778364, and 6.0032137298.

```
% Example 11
Z = zeros(n,n+2), z = zeros(1,n+2)
I = diffmat([n n+2],0), D = diffmat([n n+2],1), D2 = diffmat([n n+2],2)
lefteval = diffrow(n+2,0,-1), righteval = diffrow(n+2,0,1)
```

	$v \mapsto \varepsilon v'' + v$	u		I		u
$u \mapsto \varepsilon u'' - u$			$=$ σ		I	
$u \mapsto u(-1)$		v				v
$u \mapsto u(1)$						
	$v \mapsto v(-1)$					
	$v \mapsto v(1)$					

FIG. 12.1. *Block operator diagram for the SVD equations* (12.2)*, taking the form of a generalized eigenvalue problem, Example* 11*. Blank blocks and rows are zero. If the blocks have dimensions* $n \times (n+2)$ *after discretization and the rows have dimensions* $1 \times (n+2)$*, giving a square matrix overall of dimensions* $(2n+4) \times (2n+4)$*, full precision for the first three singular values is achieved with* $n = 22$*.*

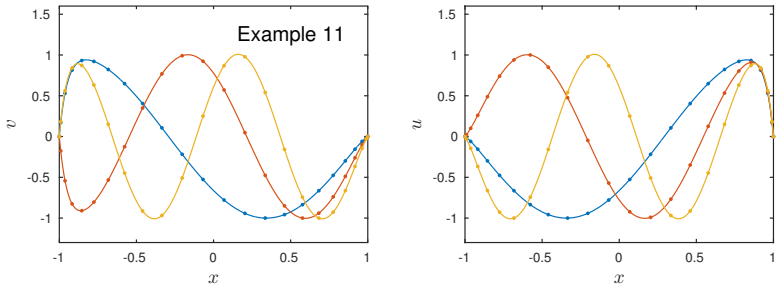

FIG. 12.2. *First three left and right singular functions* v *and* u *for the SVD problem* (12.2) *with* $\varepsilon = 0.05$ *discretized by a* $(2n+4) \times (2n+4)$ *block matrix with* $n = 22$*.*

```
A = [ Z          ep*D2+D
      ep*D2-D    Z
      lefteval   z
      righteval  z
      z          lefteval
      z          righteval ]
B = [I Z; Z I; z z; z z; z z; z z]
[UV,Lambda] = eig(A,B)
```

13. ODE-constrained optimization (Example 12). Our final example comes from the large and growing area of ODE- and PDE-constrained optimization and control [12, 13, 21]. Here we extend the problem (9.1) by considering a constraint that is not a scalar but an ODE, which will lead to a Lagrange multiplier that is not a scalar but a function.

Consider a nonlinear pendulum with angle $u(t)$ to which is applied a torque $w(t)$ over a time interval $[0, d]$,

$$f(u, w) = u'' + \sin(u) - w = 0, \quad t \in [0, d], \quad u(0) = \alpha, \quad u'(0) = \beta. \tag{13.1}$$

We wish to drive u to zero with minimal input w in the sense of minimizing $g(u, w) =$

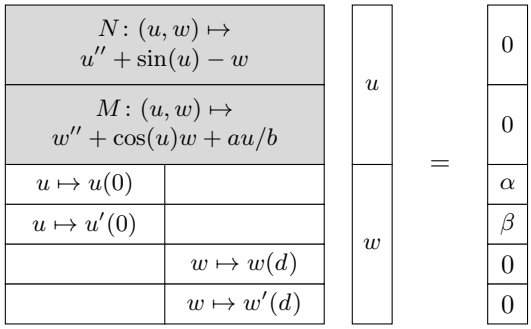

FIG. 13.1. *Nonlinear block operator diagram for control of a nonlinear pendulum, from* (13.1) *and* (13.5)*, Example* 12.

$(a/2)u^Tu + (b/2)w^Tw$, where $a, b > 0$ are constants and the inner products represent integrals over $[0, d]$. Thus the ODE-constrained optimization problem is

$$\min_{u,w} g(u,w) \quad \text{s.t.} \quad f(u,w) = 0, \quad u(0) = \alpha, \quad u'(0) = \beta.$$

The Lagrangian for this system is $\mathcal{L}(u,w,\lambda) = g(u,w) + \lambda^T f(u,w)$, where λ is a function of t, and the KKT conditions are

$$\mathcal{L}_u(u,w,\lambda) = \mathcal{L}_w(u,w,\lambda) = \mathcal{L}_\lambda(u,w,\lambda) = 0, \tag{13.2}$$

where we may calculate

$$\mathcal{L}_u = g_u^T + \lambda^T f_u, \quad \mathcal{L}_w = g_w^T + \lambda^T f_w, \quad \mathcal{L}_\lambda = f \tag{13.3}$$

and further

$$g_u = au, \quad g_w = bw, \quad f_u\colon v \mapsto v'' + \cos(u)v, \quad f_w\colon q \mapsto -q. \tag{13.4}$$

From these expressions for $\mathcal{L}_w$, g_w, and f_w we see that the equation $\mathcal{L}_w = 0$ is just $\lambda = bw$, enabling us to eliminate the unknown function λ from (13.2)–(13.4). In particular, $\mathcal{L}_u = 0$ becomes $g_u + bf_u^*w = 0$, where f_u^* denotes the adjoint operator to f_u, with adjoint boundary conditions $w(d) = w'(d) = 0$. The whole problem reduces to (13.1) together with the equations

$$w'' + \cos(u)w + au/b = 0, \quad w(d) = w'(d) = 0 \tag{13.5}$$

as shown in the block diagram of Figure 13.1 and its linearization for a Newton iteration in Figure 13.2. A solution computed with $d = 10$, $a = b = \alpha = 1$, and $\beta = 0$ appears in Figure 13.3.

14. Conclusions. Many further examples could have been added to this presentation to illustrate such features as third- or fourth-order derivatives, nonlinear boundary conditions (where rows as well as blocks are shaded and then linearized), coupled nonlinear systems and multiphysics, partial differential equations in x and t, piecewise-smooth coefficients, subdivision of domains for algorithmic reasons such as parallelization, further block formulations inspired by linear algebra such as Schur complements and least-squares problems and pseudoinverses, and more advanced

$$
\begin{array}{|c|c|}
\hline
v \mapsto \\ v'' + \cos(u)v & q \mapsto -q \\
\hline
v \mapsto av/b \\ -\sin(u)wv & q \mapsto \\ q'' + \cos(u)q \\
\hline
v \mapsto v(0) & \\
\hline
v \mapsto v'(0) & \\
\hline
 & q \mapsto q(d) \\
\hline
 & q \mapsto q'(d) \\
\hline
\end{array}
\quad
\begin{array}{|c|}
\hline
\\ \\ v \\ \\ \\
\hline
\\ \\ \\ q \\ \\ \\
\hline
\end{array}
\quad = \quad
\begin{array}{|c|}
\hline
N(u,w) \\
\hline
M(u,w) \\
\hline
0 \\
\hline
0 \\
\hline
0 \\
\hline
0 \\
\hline
\end{array}
$$

FIG. 13.2. *Linearization of Figure* 13.1 *for Newton iteration, with* $v = u^{(k)} - u^{(k+1)}$ *and* $q = w^{(k)} - w^{(k+1)}$.

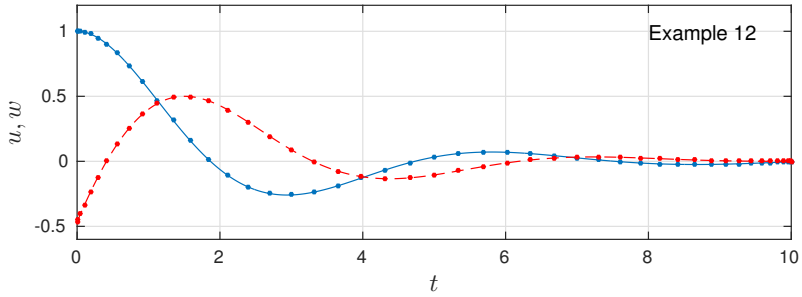

FIG. 13.3. *Solution of the nonlinear pendulum problem* (13.1), (13.5) *by* 6 *steps of Newton iteration applied to Figure* 13.2 *with* $n = 45$ *starting from the initial guess* $u = 1$, $w = 0$. *The pendulum angle* $u(t)$ *is solid and the applied torque* $w(t)$ *is dashed.*

problems of control and optimization. Problems involving initial value problems rather than BVPs can be formulated in a block fashion too, but these may be less useful in practice since one would normally solve such problems numerically by marching rather than by global matrix discretizations. As mentioned in the introduction, it is also natural to speculate about generalizations to multiple space dimensions, though we think this direction is a challenging one. Once one is in the habit of working with block operator diagrams, they begin to feel indispensable.

Acknowledgments. This article originated in discussions with Toby Driscoll of the University of Delaware, following the key paper by Driscoll and Hale [8], and in the work of Hrothgar as a graduate student at Oxford during 2013–14 under the supervision of the second author [14]. As explained in the introduction, it is an outgrowth of work of these and other members of the Chebfun team, especially Ásgeir Birkisson, which has led to the automated realization of many of these ideas in software. We thank all of these colleagues for their crucial contributions, and we thank Hrothgar additionally for providing TikZ macros for generating our block diagrams. Other colleagues have provided helpful advice including Patrick Farrell, Luc Nguyen, Mikael Slevinsky, Andy Wathen, and two very constructive anonymous referees. Example 12 on ODE-constrained optimization was developed in discussion with Birkisson and Hadrien Montanelli.

REFERENCES

[1] M. BENZI, G. H. GOLUB, AND J. LIESEN, Numerical solution of saddle point problems, *Acta Numer.* 14 (2005), 1–137.

[2] J.-P. BERRUT AND L. N. TREFETHEN, Barycentric Lagrange interpolation, *SIAM Rev.* 46 (2004), 501–517.

[3] Á. BIRKISSON AND T. A. DRISCOLL, Automatic Fréchet differentiation for the numerical solution of boundary-value problems, *ACM Trans. Math. Softw. (TOMS)*, 38 (2012), 26:1–26:29.

[4] M. BORN, W. HEISENBERG, AND P. JORDAN, Zur Quantenmechanik II, *Z. Phys.* 35 (1926), 557–615. English translation available in B. L. VAN DER WAERDEN, ed., *Sources of Quantum Mechanics,* Dover, New York, 1968.

[5] J. BOYD, *Chebyshev and Fourier Spectral Methods,* 2nd ed. (revised), Dover, 2001.

[6] C. W. CLENSHAW, The numerical solution of linear differential equations in Chebyshev series, *Math. Proc. Camb. Phil. Soc.* 53 (1957), 134–149.

[7] T. A. DRISCOLL, Automatic spectral collocation for integral, integro-differential, and integrally reformulated differential equations, *J. Comp. Phys.* 229 (2010), 5980–5998.

[8] T. A. DRISCOLL AND N. HALE, Rectangular spectral collocation, *IMA J. Numer. Anal.* 36 (2016), 108–132.

[9] T. A. DRISCOLL, N. HALE, AND L. N. TREFETHEN, eds., *Chebfun User's Guide,* Pafnuty Publications, Oxford, 2014; see also `www.chebfun.org`.

[10] G. H. GOLUB AND C. F. VAN LOAN, *Matrix Computations,* 4th ed., Johns Hopkins, 2013.

[11] L. GREENGARD, Spectral integration and two-point boundary value problems, *SIAM J. Numer. Anal.* 28 (1991), 1071–1080.

[12] M. D. GUNZBURGER, *Perspectives in Flow Control and Optimization,* SIAM, 2003.

[13] M. HINZE, R. PINNAU, M. ULBRICH, AND S. ULBRICH, *Optimization with PDE Constraints,* Springer, 2009.

[14] HROTHGAR, *Block Operators and Continuous Adjoint Methods,* transfer thesis, Mathematical Institute, University of Oxford, November 2014.

[15] T. KATO, *Perturbation Theory for Linear Operators,* 2nd ed., Springer-Verlag, 1976.

[16] S. OLVER AND A. TOWNSEND, A practical framework for infinite-dimensional linear algebra, *Proceedings of the 1st Workshop for High Performance Technical Computing in Dynamic Languages,* IEEE Press, 2014, pp. 57–62.

[17] S. SMALE, An infinite dimensional version of Sard's theorem, *Amer. J. Math.* 87 (1965), 861–866.

[18] L. N. TREFETHEN, *Spectral Methods in MATLAB,* SIAM, 2000.

[19] L. N. TREFETHEN, *Approximation Theory and Approximation Practice,* SIAM, 2013.

[20] C. TRETTER, *Spectral Theory of Block Operator Matrices and Applications,* Imperial College Press, 2008.

[21] F. TRÖLTZSCH, *Optimal Control of Partial Differential Equations: Theory, Methods and Applications,* AMS, 2010.

[22] D. S. WATKINS, *Fundamentals of Matrix Computations,* 3rd ed., Wiley, 2010.

[23] G. B. WRIGHT, M. JAVED, H. MONTANELLI, AND L. N. TREFETHEN, Extension of Chebfun to periodic functions, *SIAM J. Sci. Comp.* 37 (2015), C554–C573.

[24] K. XU AND N. HALE, Explicit construction of rectangular differentiation matrices, *IMA J. Numer. Anal.* 36 (2016), 618–632.

[25] E. ZEIDLER, *Nonlinear Functional Analysis and its Applications, I,* Springer-Verlag, 1986.