

# 1 Weeks 14 and 15: Nov 25 through Dec 4, 2003

Finally, we will present the derivation of fast methods for numerical solution of integral equations corresponding to linear two-point boundary value problems (BVPs)

$$\begin{aligned}y' &= P(t)y(t) + f(t) & a < t < b, \\Ay(a) + By(b) &= g.\end{aligned}$$

Useful references for this material include [GR91, SR94].

## 2 The integral equation

The integral equation is based on a “background problem”

$$\begin{aligned}y' &= P_0(t)y(t) + \sigma(t) & a < t < b, \\Ay(a) + By(b) &= g.\end{aligned}$$

with a known fundamental matrix  $Y_0$  and Green function  $G_0$ . Most commonly we take  $P_0 = 0$  so  $Y_0 = I$  and

$$G_0(t, s) = \begin{cases} D^{-1}A & a < s < t < b \\ -D^{-1}B & a < t < s < b \end{cases}$$

The integral equation is then

$$\sigma(t) + (P_0(t) - P(t)) \int_a^b G_0(t, s)\sigma(s)ds = f(t) - (P_0(t) - P(t))Y_0(t)D_0^{-1}g.$$

Because of the semi-separable form of  $G_0$ , the equation can also be written as

$$\begin{aligned}\sigma(t) &+ (P_0(t) - P(t))Y_0(t)D_0^{-1}AY_0(a) \int_a^t Y_0(s)^{-1}\sigma(s)ds \\ &- (P_0(t) - P(t))Y_0(t)D_0^{-1}BY_0(b) \int_t^b Y_0(s)^{-1}\sigma(s)ds \\ &= f(t) - (P_0(t) - P(t))Y_0(t)D_0^{-1}g = r(t)\end{aligned}$$

or

$$\mu(t) + \mathcal{A}(t) \int_a^t \mu(s)ds - \mathcal{B}(t) \int_t^b \mu(s)ds = \rho(t)$$

where we have multiplied through by  $Y_0^{-1}(t)$  and defined  $\mu(t) = Y_0^{-1}(t)\sigma(t)$ . Of course  $\sigma = \mu$  for the simplest background problem with  $P_0 = 0$ . Once this integral equation is solved, the solution  $y$  can be evaluated from either  $\sigma$  or  $\mu$  by the Green formula.

The spectral element method decomposes the range of integration  $[a, b]$  into subintervals  $[t_n, t_{n+1}]$  and uses a fixed high-order Legendre rule on each subinterval. Define  $Nm$  compound Legendre points on  $[a, b]$  by

$$h = (b - a)/N, \quad s_{Im+i} = a + h(I + t_i), \quad v_{Im+i} = hw_i$$

for  $I = 0, \dots, N - 1, i = 1, \dots, m$ , where  $t_i$  and  $w_i$  are the Legendre points and weights on  $[0, 1]$ . Then we can compute indefinite integrals to high accuracy by

$$\begin{aligned} \int_a^{s_{Im+i}} f(s) ds &= \sum_{J=0}^{I-1} \int_{a+Jh}^{a+(J+1)h} f(s) ds + \int_{a+Ih}^{a+Ih+ht_i} f(s) ds \\ &= \sum_{J=0}^{I-1} \sum_{j=1}^m v_j f(s_{Jm+j}) + \sum_{j=1}^m h S_{ij} f(s_{Jm+j}) + O(h^m) \\ &= \sum_{J=0}^I \sum_{j=1}^m \mathcal{S}_{Im+i, Jm+j} f(s_{Jm+j}) + O(h^m). \end{aligned}$$

Here

$$\mathcal{S}_{Im+i, Jm+j} = \begin{cases} 0 & J > I \\ h S_{ij} & J = I \\ h w_j & J < I \end{cases}$$

and

$$\mathcal{T}_{Im+i, Jm+j} = \begin{cases} h w_j & J > I \\ h T_{ij} & J = I \\ 0 & J < I \end{cases}$$

is the corresponding matrix for integrating from  $t$  up to  $b$ , with  $T_{ij} = S_{m+1-i, m+1-j}$  the local spectral integration matrix from  $t_i$  up to 1. After we discretize the integral equation with these points and the spectral integration matrix, we get a matrix-vector equation  $Ms = r$  where

$$\begin{aligned} s &= (\sigma(s_1), \sigma(s_2), \dots, \sigma(s_{Nm}))^T, \\ r &= (r(s_1), r(s_2), \dots, r(s_{Nm}))^T, \end{aligned}$$

and each  $p \times p$  block of the  $Nmp \times Nmp$  matrix  $M$  is given by

$$\begin{aligned} M_{Im+i, Jm+j} &= \delta_{IJ} \delta_{ij} I_p + (P_0(s_{Im+i}) - P(s_{Im+i})) Y_0(s_{Im+i}) D_0^{-1} \\ &\quad (A Y_0(a) \mathcal{S}_{Im+i, Jm+j} - B Y_0(b) \mathcal{T}_{Im+i, Jm+j}) Y_0^{-1}(s_{Jm+j}). \end{aligned}$$

For the simplest background problem with  $P_0 = 0$ , this simplifies to

$$M_{Im+i, Jm+j} = \delta_{IJ} \delta_{ij} I_p - P(s_{Im+i}) D_0^{-1} (A \mathcal{S}_{Im+i, Jm+j} - B \mathcal{T}_{Im+i, Jm+j}).$$

### 3 Fast multiplication

For simplicity, we explain fast solution methods first for the transformed variable  $\mu$  which satisfies

$$\mu(t) + \mathcal{A}(t) \int_a^t \mu(s) ds - \mathcal{B}(t) \int_t^b \mu(s) ds = \rho(t).$$

Discretization gives  $\mathcal{C}\mu = \rho$  where

$$\mathcal{C}_{Im+i, Jm+j} = \delta_{IJ} \delta_{ij} I_p + \mathcal{A}(s_{Im+i}) \mathcal{S}_{Im+i, Jm+j} - \mathcal{B}(s_{Im+i}) \mathcal{T}_{Im+i, Jm+j}.$$

The key feature of the matrix  $\mathcal{C}$  is the low rank of all the off-diagonal  $mp \times mp$  blocks such as

$$\mathcal{A}(s_{Im+i}) \mathcal{S}_{Im+i, Jm+j} = \mathcal{A}(s_{Im+i}) h w_j$$

for fixed  $J < I$ . Each of these blocks is a “tensor product” of one matrix which only depends on  $I$  and  $i$  with a number which only depends on  $j$ . Such a block can be applied to the corresponding block of a vector by taking the dot product of the vector with a vector of weights  $w$  then multiplying each size- $p$  block of the result by a  $p \times p$  matrix  $\mathcal{A}(s_{Im+i})$ . The cost of such an operation is not the  $O(m^2 p^2)$  one would expect for multiplying an  $mp \times mp$  block times a vector of length  $mp$ , but rather  $O(mp^2)$ . This factor of  $m$  is significant for high-order discretization with  $m = 32$ , for example, and allows us to apply the matrix to a vector much faster than one would expect from its size. In fact, because the indefinite integral is an additive function of the interval, the matrix can be applied to a vector in just  $O(mNp^2)$  work instead of  $O(m^2 N^2 p^2)$ , a tremendous savings. This potential for efficient application—together with the bounded condition numbers of these matrices as  $h \rightarrow 0$ —encourages us to solve these integral equation by iterative black-box solver such as GMRES [SS86]. Such methods would normally offer no advantage for these problems because the matrices are small and dense, with only an occasional zero element, so the essential realization is that the special structure of these matrices allows the matrix-vector product to be formed fast.

More precisely, we can compute the matrix-vector product  $\mathcal{C}x$  by the following steps:

1. Evaluate the indefinite integrals at interval endpoints

$$\Omega_I = \sum_{J=0}^{I-1} \sum_{j=1}^m h w_j x_{Jm+j} = \Omega_{I-1} + \sum_{j=1}^m h w_j x_{Im+j}$$

for  $1 \leq I \leq N$ , at a total cost of  $O(Nmp)$  flops.

2. Evaluate the local indefinite integrals

$$y_{Im+i} = h \sum_{j=1}^m S_{ij} x_{Im+j}$$

and

$$z_{Im+i} = h \sum_{j=1}^m T_{ij} x_{Im+j}$$

for  $0 \leq I \leq N-1$  and  $1 \leq i \leq m$ , at a total cost of  $O(Nm^2p)$  flops.

3. Evaluate each  $p$ -block of the product  $\mathcal{C}x$  by

$$(\mathcal{C}x)_{Im+i} = x_{Im+i} + \mathcal{A}(s_{Im+i})(\Omega_I + y_{Im+i}) - \mathcal{B}(s_{Im+i})(\Omega_N - \Omega_I + z_{Im+i})$$

for  $0 \leq I \leq N-1$  and  $1 \leq i \leq m$ , at a total cost of  $O(Nmp^2)$  flops.

Summing up, the total cost is  $O(N(mp^2 + m^2p))$  flops. Since the condition number of the matrix is bounded, folklore suggests that GMRES will take a fixed number of steps to converge even for very small  $h$ , so the linear system can be solved for  $\mu$  in total work  $O(N(mp^2 + m^2p))$ . This is much less than the  $O(N^3m^3p^3)$  required by Gaussian elimination if the matrix is large. The same analysis works for the more complicated system satisfied by  $\sigma$ , which we might prefer in practice because multiplication by  $Y_0$  may amplify ill-conditioning if  $P_0 \neq 0$ .

## 4 Fast direct methods

It is possible to take even more advantage of the low-rank off-diagonal blocks of this matrix, to solve the integral equation directly without iteration. This is desirable because when the BVP is somewhat ill-conditioned then GMRES may require several hundred iterations to converge, reducing its advantage over Gaussian elimination. This fast direct solution strategy happens because of the analytical structure of integral equations for two-point BVPs, which connect solutions on different intervals by low-rank operators. Let's split our interval  $[a, b]$  into two intervals  $L = [a, c]$  and  $R = [c, b]$ , and break up the integral operator accordingly. Then the equation

$$\mu(t) + A(t) \int_a^t \mu(s) ds - B(t) \int_t^b \mu(s) ds = \rho(t) \quad a < t < b$$

becomes a pair of integral equations

$$\begin{aligned} \mu_L(t) + A(t) \int_a^t \mu_L(s) ds - B(t) \int_t^c \mu_L(s) ds - B(t) \int_c^b \mu_R(s) ds &= \rho_L(t) & a < t < c \\ \mu_R(t) + A(t) \int_c^t \mu_R(s) ds + A(t) \int_a^c \mu_L(s) ds - B(t) \int_t^b \mu_R(s) ds &= \rho_R(t) & c < t < b, \end{aligned}$$

where subscripts indicate domains. We can write this symbolically as

$$\begin{aligned} \mu_L + \mathcal{A}_{LL}\mu_L - \mathcal{B}_{LL}\mu_L - \mathcal{B}_{LR}\mu_R &= \rho_L \\ \mu_R + \mathcal{A}_{RR}\mu_R - \mathcal{B}_{RR}\mu_R + \mathcal{A}_{RL}\mu_L &= \rho_R \end{aligned} \tag{1}$$

or

$$\begin{bmatrix} I + \mathcal{A}_{LL} - \mathcal{B}_{LL} & -\mathcal{B}_{LR} \\ \mathcal{A}_{RL} & I + \mathcal{A}_{RR} - \mathcal{B}_{RR} \end{bmatrix} \begin{bmatrix} \mu_L \\ \mu_R \end{bmatrix} = \begin{bmatrix} \rho_L \\ \rho_R \end{bmatrix}.$$

This structure is extremely interesting: Each density satisfies an integral equation of exactly the type we started out with on each interval, plus the two separate integral equations are connected by low-rank operators such as

$$\mathcal{B}_{LR}\mu_R = B(t) \int_c^b \mu_R(s) ds$$

and

$$\mathcal{A}_{RL}\mu_L = A(t) \int_a^c \mu_L(s) ds.$$

These operators integrate  $\mu_L$  or  $\mu_R$  over a *fixed* interval, yielding a constant  $p$ -vector, then multiply it by a known matrix function of  $t$ . This is very similar to a low-rank matrix  $y = (uv^T)x$  which takes the dot product of  $x$  with  $v$  and then multiplies the resulting scalar by the vector  $u$ . As we saw above, such low-rank matrices can be applied fast to arbitrary vectors. However, what we want now is to *invert* perturbations of invertible matrices (of operators) by low-rank off-diagonal blocks.

This is done by variations and applications of the so-called ‘‘Sherman–Woodbury–Morrison formula’’

$$(I + uv^T)^{-1} = I + \alpha v u^T, \quad \alpha = \frac{-1}{1 + v^T u}$$

which holds iff  $v^T u \neq -1$ . This formula shows that when we solve the linear system

$$x + (v^T x)u = b,$$

we need only represent  $x$  as a linear combination of  $b$  and  $u$ , say  $\beta b + \omega u$ , and find the coefficients from

$$\beta b + \omega u + (\beta v^T b + \omega v^T u)u = b.$$

We find  $\beta = 1$  and

$$\omega = \frac{-1}{1 + v^T u} v^T b,$$

which leads to

$$x = b - \frac{v^T u}{1 + v^T u} v^T b = \left( I - \frac{uv^T}{1 + v^T u} \right) b$$

as advertised.

A similar technique works with low-rank integral equations. For example, we can solve a second-kind Fredholm equation with a rank-one kernel

$$\sigma(t) + \int_a^b u(t)v(s)\sigma(s) ds = f(t)$$

by seeking a solution  $\sigma$  as a linear combination of  $f$  and  $u$ , say  $\beta f + \omega u$ . Then substituting in gives

$$\beta f(t) + \omega u(t) + \int_a^b u(t)v(s)(\beta f(s) + \omega u(s))ds = f(t)$$

or

$$\beta f(t) + (\omega + \int_a^b v(s)(\beta f(s) + \omega u(s))ds)u(t) = f(t).$$

As in the finite-dimensional case, we find  $\beta = 1$  and

$$(1 + \int_a^b v(s)u(s)ds)\omega = \int_a^b v(s)f(s)ds$$

so if  $\int_a^b v(s)u(s)ds \neq -1$  then the inverse operator is given by

$$\sigma(t) = f(t) - \frac{1}{1 + \int_a^b v(s)u(s)ds} \int_a^b v(s)f(s)ds.$$

These formulas can be extended to solve low-rank perturbations of other invertible operators than the identity, because

$$\begin{aligned} (A + uv^T)^{-1} &= (A(I + A^{-1}uv^T))^{-1} \\ &= (I + (A^{-1}u)v^T)^{-1}A^{-1} \\ &= (I - \frac{1}{1 + v^T A^{-1}u}(A^{-1}u)v^T)A^{-1} \\ &= A^{-1} - \frac{1}{1 + v^T A^{-1}u}A^{-1}u(A^{-T}v)^T. \end{aligned} \quad (2)$$

Thus perturbing an invertible matrix by a rank-one matrix perturbs its inverse with a different rank-one matrix. In the language of linear systems rather than (deprecated) matrix inverses, solving a system after a rank-one change requires two solves of the original system: First, compute  $w = A^{-1}u$  and  $\alpha = \frac{1}{1 + v^T w}$  (in a preprocessing step independent of the right-hand side), then solve  $(A + uv^T)x = b$  by solving  $Ay = b$  for  $y = A^{-1}b$  and apply the rank-one matrix  $B = I - \alpha wv^T$  to get  $x = By = y - \alpha v^T y w$ .

All these formulas generalize immediately to low-rank rather than rank-one perturbations of invertible matrices or operators: if  $U$  and  $V$  are  $n \times k$  matrices with small width  $k$  compared to their height  $n$  and  $A$  is invertible, then  $A + UV^T$  is invertible and its inverse is

$$(A + UV^T)^{-1} = A^{-1} - A^{-1}U(I + V^T A^{-1}U)^{-1}V^T A^{-1}$$

iff the  $k \times k$  matrix  $(I + V^T A^{-1}U)$  is invertible. This allows us to find exact solutions (in terms of integrals) for second-kind Fredholm equations with “finite-rank” kernels

$$K(t, s) = \sum_{i=1}^p u_i(t)v_i(s),$$

where we assume for simplicity that  $u_i$  and  $v_i$  form linearly independent sets of functions. Seek the solution  $\sigma$  of

$$\sigma(t) + \int_a^b K(t, s)\sigma(s)ds = f(t)$$

as a linear combination of  $f$  and the  $u_i$ 's:

$$\sigma(t) = \beta f(t) + \sum_{i=1}^p \omega_i u_i(t)$$

will hold iff

$$\beta f(t) + \sum_{i=1}^p \omega_i u_i(t) + \sum_{i=1}^p u_i(t) \int_a^b v_i(s)(\beta f(s) + \sum_{j=1}^p \omega_j u_j(s))ds = f(t).$$

This will hold if  $\beta = 1$  and

$$\sum_{i=1}^p u_i(t) \left[ \omega_i + \sum_{j=1}^p \int_a^b v_i(s)u_j(s)ds\omega_j \right] = - \sum_{i=1}^p u_i(t) \int_a^b v_i(s)f(s)ds.$$

Equating coefficients, we see that it is enough to satisfy  $p$  equations

$$\omega_i + \sum_{j=1}^p \int_a^b v_i(s)u_j(s)ds\omega_j = - \int_a^b v_i(s)f(s)ds$$

in the  $p$  unknowns  $\omega_i$ . Let  $(V^T U)_{ij} = \int_a^b v_i(s)u_j(s)ds$  and  $b_i = - \int_a^b v_i(s)f(s)ds$ ; then

$$\omega = (I + V^T U)^{-1} b$$

gives the coefficients of the solution  $\sigma$ . The formula for  $\sigma$  then corresponds exactly to the generalized Sherman-Morrison-Woodbury formula above.

Returning to our integral equation for the two-point BVP, we assume that the diagonal operators  $I + \mathcal{A}_{LL} - \mathcal{B}_{LL}$  and  $I + \mathcal{A}_{RR} - \mathcal{B}_{RR}$  are invertible. Then multiplying by the  $2 \times 2$  diagonal matrix with their inverses on the diagonal, or equivalently solving the integral equation on each interval separately, gives a  $2 \times 2$  matrix of operators

$$\begin{bmatrix} I & -\mathcal{B}'_{LR} \\ \mathcal{A}'_{RL} & I \end{bmatrix}$$

where the primed low-rank off-diagonal operators operate on functions by integrating them over the whole domain and multiplying the resulting constant  $p$ -vector by some matrix function of  $t$  found by solving the integral equation on each half-interval. An application of the generalized SMW formula gives an exact formula for the solution of the integral equation in terms of its solutions on each half-interval, which leads to fast recursive algorithms.

Explicitly, we have

$$\begin{aligned}\mathcal{B}_{LR}\mu_R &= B(t) \int_c^b \mu_R(s) ds \\ \mathcal{A}_{RL}\mu_L &= A(t) \int_a^c \mu_L(s) ds.\end{aligned}$$

Define matrix functions  $C_L(t)$  and  $\chi_L(t)$  on  $L$  and  $C_R(t)$  and  $\chi_R(t)$  on  $R$  by solving the integral equations

$$\begin{aligned}(I + \mathcal{A}_{LL} - \mathcal{B}_{LL})C_L &= -B(t) && \text{on } L \\ (I + \mathcal{A}_{LL} - \mathcal{B}_{LL})\chi_L &= \rho_L(t) && \text{on } L \\ (I + \mathcal{A}_{RR} - \mathcal{B}_{RR})C_R &= A(t) && \text{on } R \\ (I + \mathcal{A}_{RR} - \mathcal{B}_{RR})\chi_R &= \rho_R(t) && \text{on } R,\end{aligned}\tag{3}$$

so that our integral equations are equivalent to

$$\begin{aligned}\mu_L + C_L(t) \int_c^b \mu_R(s) ds &= \chi_L(t) && \text{on } L \\ \mu_R + C_R(t) \int_a^c \mu_L(s) ds &= \chi_R(t) && \text{on } R.\end{aligned}$$

Integrate each equation, multiply it by the  $C$  matrix appearing in the other equation, and subtract to get a pair of low-rank perturbations of the identity operator:

$$\begin{aligned}\mu_L(t) - C_L(t) \int_c^b C_R(s) ds \int_a^c \mu_L(s) ds &= \chi_L(t) - C_L(t) \int_c^b \chi_R(s) ds \\ \mu_R(t) - C_R(t) \int_a^c C_L(s) ds \int_c^b \mu_R(s) ds &= \chi_R(t) - C_R(t) \int_a^c \chi_L(s) ds.\end{aligned}$$

Solve them by the usual rank-one approach: assume solutions of the form

$$\begin{aligned}\mu_L(t) &= \chi_L(t) - C_L(t) \int \chi_R(s) ds + C_{LR}(t)\omega_L \\ \mu_R(t) &= \chi_R(t) - C_R(t) \int \chi_L(s) ds + C_{RL}(t)\omega_R\end{aligned}$$

where  $C_{LR} = C_L(t) \int C_R$  and  $C_{RL}(t) = C_R(t) \int C_L$ , and plug in to the equations to get

$$\begin{aligned}(I - \int C_{LR})\omega_L &= \int \left( \chi_L - C_L \int \chi_R \right) \\ (I - \int C_{RL})\omega_R &= \int \left( \chi_R - C_R \int \chi_L \right).\end{aligned}\tag{4}$$

Solve these condensed  $p \times p$  linear systems for  $\omega_L$  and  $\omega_R$ , and  $\mu_L$  and  $\mu_R$  are found. This allows us to solve the integral equation on any interval if we

can solve it on both parts of the interval, at a small additional cost. Now we can build several fast algorithms for solution of the integral equation.

The simplest approach is a bottom-up method: solve on each of  $N = 2^L$  intervals with  $m$  Legendre points each for the quantities  $\chi$  and  $C$  on level  $L$ , at a total cost of  $O(Nm^3p^3)$  if Gaussian elimination is used. Then recursively build the solutions  $\chi$  and  $C$  on level  $l = L - 1, \dots, 0$  by piecing together adjacent pairs of intervals at cost  $O(2^l p^3 m^3)$  on level  $l$ , total cost  $O(Nm^3p^3)$ .

The more complex technique relies on the representation

$$\begin{aligned}\mu_L(t) &= \chi_L(t) - C_L(t) \int \chi_R(s) ds + C_{LR}(t) \omega_L \\ \mu_R(t) &= \chi_R(t) - C_R(t) \int \chi_L(s) ds + C_{RL}(t) \omega_R\end{aligned}$$

for the solution on each subinterval, and particularly on the existence of the vectors  $\omega$ .

Now return to

$$\begin{aligned}K_L \mu_L - B(t) \bar{\mu}_R &= \rho_L, \\ K_R \mu_R + A(t) \bar{\mu}_L &= \rho_R\end{aligned}$$

where bars denote integration over the domain:

$$\begin{aligned}\bar{\mu}_R &= \int_R \mu_R(s) ds, \\ \bar{\mu}_L &= \int_L \mu_L(s) ds.\end{aligned}$$

These equations can be solved by a two-step process. First, we multiply through by the inverses of  $K_L$  and  $K_R$ ; in other words, we solve the integral equation (recursively) on  $L$  and  $R$ , ignoring the coupling between  $L$  and  $R$  which influences the solution  $\mu$ . Second, we eliminate each variable  $\mu_L$  and  $\mu_R$  from the equation determining the other, obtaining a low-rank perturbation of the identity which is easy to invert. Since we have to solve matrix as well as vector equations, we formulate the integral equation as part of a larger problem which will then be reproduced by each subdivision:

$$\begin{aligned}K\mu &= \rho(t) && \text{on } L \cup R, \\ K\alpha &= A(t) && \text{on } L \cup R, \\ K\beta &= -B(t) && \text{on } L \cup R.\end{aligned}$$

Applying the inverses on each interval  $L$  and  $R$  separately, we define  $\chi_L, \chi_R, \alpha_L, \alpha_R, \beta_L$ , and  $\beta_R$  to be the solutions of the integral equations

$$\begin{aligned}K_L \chi_L &= \rho && \text{on } L, \\ K_L \alpha_L &= A && \text{on } L, \\ K_L \beta_L &= -B && \text{on } L, \\ K_R \chi_R &= \rho && \text{on } R, \\ K_R \alpha_R &= A && \text{on } R, \\ K_R \beta_R &= -B && \text{on } R.\end{aligned}$$

Thus our coupled system becomes

$$\begin{aligned} K_L \mu_L + (K_L \beta_L) \bar{\mu}_R &= K_L \chi_L & \text{on } L, \\ K_R \mu_R + (K_R \alpha_R) \bar{\mu}_L &= K_R \chi_R & \text{on } R. \end{aligned}$$

Canceling the (assumed invertible) operators  $K_L$  and  $K_R$  from each equation gives

$$\begin{aligned} \mu_L + \beta_L \bar{\mu}_R &= \chi_L & \text{on } L, \\ \mu_R + \alpha_R \bar{\mu}_L &= \chi_R & \text{on } R. \end{aligned}$$

Now integrating each equation, multiplying by the coefficient of the other, and subtracting gives a pair of decoupled equations

$$\begin{aligned} \mu_L - \beta_L \bar{\alpha}_R \bar{\mu}_L &= \chi_L - \beta_L \bar{\chi}_R & \text{on } L, \\ \mu_R - \alpha_R \bar{\beta}_L \bar{\mu}_R &= \chi_R - \alpha_R \bar{\chi}_L & \text{on } R, \end{aligned}$$

which are clearly recognizable as low-rank perturbations of the identity operator. Representing their solutions as equal to the right-hand side plus some constant vector  $\omega$  multiplied by the matrix coefficient, corresponding to the ansatz  $x = b + \omega u$  for the solution of  $x + uv^T x = b$ , yields

$$\begin{aligned} \mu_L &= \chi_L - \beta_L \bar{\chi}_R + \beta_L \bar{\alpha}_R \omega_L, \\ \mu_R &= \chi_R - \alpha_R \bar{\chi}_L + \alpha_R \bar{\beta}_L \omega_R. \end{aligned}$$

Plugging into the equations shows that the  $\omega$ 's satisfy a pair of  $p \times p$  linear systems

$$\begin{aligned} (I - \bar{\beta}_L \bar{\alpha}_R) \omega_L &= \bar{\chi}_L - \bar{\beta}_L \bar{\chi}_R, \\ (I - \bar{\alpha}_R \bar{\beta}_L) \omega_R &= \bar{\chi}_R - \bar{\alpha}_R \bar{\chi}_L. \end{aligned}$$

Assuming the matrices of these systems are invertible, we have

$$\begin{aligned} \mu_L &= \chi_L - \beta_L \bar{\chi}_R + \beta_L \bar{\alpha}_R (I - \bar{\beta}_L \bar{\alpha}_R)^{-1} (\bar{\chi}_L - \bar{\beta}_L \bar{\chi}_R) \\ &= (I + \beta_L \bar{\alpha}_R (I - \bar{\beta}_L \bar{\alpha}_R)^{-1} E) (\chi_L - \beta_L \bar{\chi}_R) \\ &= M_L (\chi_L - \beta_L \bar{\chi}_R), \\ \mu_R &= \chi_R - \alpha_R \bar{\chi}_L + \alpha_R \bar{\beta}_L (I - \bar{\alpha}_R \bar{\beta}_L)^{-1} (\bar{\chi}_R - \bar{\alpha}_R \bar{\chi}_L) \\ &= (I + \alpha_R \bar{\beta}_L (I - \bar{\alpha}_R \bar{\beta}_L)^{-1} E) (\chi_R - \alpha_R \bar{\chi}_L) \\ &= M_R (\chi_R - \alpha_R \bar{\chi}_L) \end{aligned}$$

where we have defined the integration operator  $E \mu_L = \bar{\mu}_L$  and named the merge operators  $M_L$  and  $M_R$  for future use. The merge operators construct the solution  $\mu$  on each interval from the solutions of problems on the sub-intervals. Thus the matrices  $\alpha$  and  $\beta$  on the whole interval  $L \cup R$  can be constructed by the formulas

$$\begin{aligned} \alpha &= \begin{cases} M_L (\alpha_L - \beta_L \bar{\alpha}_R) & \text{on } L \\ M_R (\alpha_R - \alpha_R \bar{\alpha}_L) & \text{on } R \end{cases} \\ \beta &= \begin{cases} M_L (\beta_L - \beta_L \bar{\beta}_R) & \text{on } L \\ M_R (\beta_R - \alpha_R \bar{\beta}_L) & \text{on } R \end{cases} \end{aligned}$$

These formulas can be applied recursively, yielding a fast recursive algorithm for the discrete problem as follows.

Suppose we discretize the interval  $[a, b]$  with  $N = 2^{l+1}$  intervals, putting  $m$  Legendre points on each, and there are  $p$  ODEs in our two-point BVP. Then each merge operator  $M_L$  or  $M_R$  can be applied to a function on  $2^l$  of the subintervals, united into a single  $L$  or  $R$ , in  $O(2^l m p^2) + O(p^3)$  work. The first term is due to integration and  $p \times p$  matrix-vector multiplication at each of the  $m$  points, while the second (and smaller) term is due to inverting (or factorizing) the  $p \times p$  matrices on each of  $L$  and  $R$ . Hence the work  $W_{l+1}$  with  $N$  subintervals satisfies

$$W_{l+1} = 2W_l + 2^l m p^3$$

since the merge operators must be applied  $4p + 2$  times, once to each vector  $\chi_L$  and  $\chi_R$  and once to each column of the  $p \times p$  matrix functions  $\alpha_L$ ,  $\alpha_R$ ,  $\beta_L$ , and  $\beta_R$ . Clearly  $W_0 = O(m^3 p^3)$  if a direct method like Gaussian elimination or QR factorization is used on the leaf intervals, though this estimate could be improved considerably if the special structure of the lowest-level  $mp \times mp$  matrices is taken into account. Since the recursion

$$W_{l+1} = 2W_l + 2^{l+1} B, \quad W_0 = A$$

has solution

$$W_l = 2^l (A + lB),$$

we see that recursive solution of the two-point BVP costs

$$W_l = O(N m^3 p^3) + O(N \log N m p^3).$$

Since  $Nm$  is the number of points where we obtain the solution, this estimate is not too bad. It compares well with the brute-force global Gaussian elimination approach which costs  $O(N^3 m^3 p^3)$  when large numbers of intervals are needed.

This approach can be made adaptive [LG97] and improved on the lowest level; codes may soon be available. Numerical results are excellent (see [GR91, SR94]).

## References

- [GR91] L. Greengard and V. Rokhlin. On the numerical solution of two-point boundary value problems. *Comm. Pure Appl. Math.*, 44(4):419–452, 1991.
- [LG97] J.-Y. Lee and L. Greengard. A fast adaptive numerical method for two-point boundary value problems. *SIAM J. Sci. Comput.*, 18:403–429, 1997.

- [SR94] Page Starr and Vladimir Rokhlin. On the numerical solution of two-point boundary value problems. II. *Comm. Pure Appl. Math.*, 47(8):1117–1159, 1994.
- [SS86] Y. Saad and M. R. Schultz. GMRES: A generalized minimum residual method for solving nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.*, 7:856–869, 1986.