Analyzing a Projection Method with Maple

Heinz H. Bauschke and Robert M. Corless†*

Abstract: We prove the following result on an asynchronous projection method:

Let X be a Hilbert space and U, V be two closed subspaces with corresponding orthogonal projections A, B. Fix two points $x_0, x_1 \in X$ and define the sequence (x_n) by

$$x_n := \tfrac{1}{2}Bx_{n-1} + \tfrac{1}{2}Ax_{n-2}, \quad \forall n \geq 2.$$

Then (x_n) converges to $C(\frac{1}{3}x_0 + \frac{2}{3}x_1)$, where C is the orthogonal projection onto $U \cap V$.

The proof is an interesting blend of combinatorics and analysis; the combinatorial part is done with Maple.

Introduction

SETTING. Throughout this paper, we assume the following:

- X is a Hilbert space;
- U, V are closed subspaces with orthogonal projections A, B;
- C is the orthogonal projection onto $U \cap V$.

(Recall that a *Hilbert space* is a complete inner product space such as $\mathbb{R}^n$. For basic properties of orthogonal projections see [4] or almost any other text on Functional Analysis.) Formulated in our setting, the *convex feasibility problem* consists of finding a point in the intersection of U and V. This problem, which occurs frequently in applications (see [1, 2] and the many references therein), can be solved iteratively provided the orthogonal projections A and B are computable:

Fact 1 (von Neumann, 1933) Let $y_0 \in X$ and generate *the sequence of alternating projections*

$$(y_0, Ay_0, BAy_0, ABAy_0, BABAy_0, ABABAy_0, \ldots).$$

Then (y_n) converges to Cy_0.

von Neumann's result is remarkable because

- the sequence (y_n) converges *in norm*; something one cannot take for granted in infinite-dimensional spaces.
- the limit is *independent of the order of the subspaces* in the sense that the sequence

  $$(y_0, By_0, ABy_0, BABy_0, \ldots)$$

  also converges to Cy_0.
- the limit of (y_n) is not only a solution of the convex feasibility problem but also the solution of a more ambitious *best approximation problem.*

*Department of Combinatorics & Optimization, University of Waterloo, Waterloo, Ontario N2L 3G1, Canada. bauschke@cecm.sfu.ca

†Department of Applied Mathematics, University of Western Ontario, London, Ontario N6A 5B7, Canada. rcorless@uwo.ca

von Neumann's method is a *sequential* method — in contrast to a *parallel* method such as

$$y_n := \tfrac{1}{2}By_{n-1} + \tfrac{1}{2}Ay_{n-1},$$

where the two operators A and B can simultaneously work on computing Ay_{n-1} and By_{n-1}. Both methods fall under the larger umbrella of *projection methods*, which derive the new iterate y_n by the application of a *well-behaved* map (which could even depend on n) to the immediate predecessor y_{n-1}. In a related though different context, Chazan and Miranker [3] (see Strikwerda's [12] for a recent reference) suggested the even more general *asynchronous* (sometimes also called "*chaotic*") projection methods where the new iterate y_n depends not only on y_{n-1} and n but possibly also on *all* previous iterates $y_{n-2}, \ldots, y_0$.

The aim of this paper is to analyze the following prototype of an asynchronous projection method in some detail:

$$\boxed{x_0,\ x_1 \in X. \qquad x_n := \tfrac{1}{2}Bx_{n-1} + \tfrac{1}{2}Ax_{n-2}, \quad \forall n \geq 2.}$$

The Computer Algebra System Maple is shown to be very useful for investigating this iteration.

The paper is organized as follows. Relevant extracts from the Maple session are presented in Section 2; in particular, it contains the representation and subsequent analysis of the iteration. In Section 3, we combine the combinatorial information gained in Section 2 with von Neumann's result and an extension of a summability result by Toeplitz in order to prove that the sequence (x_n) converges to $C(\frac{1}{3}x_0 + \frac{2}{3}x_1)$. The computability of orthogonal projections in Euclidean spaces via the Moore/Penrose inverse is discussed in Section 4.

The combinatorial part: done with Maple

If one writes down some iterates by hand, using the basic facts $A^2 = A$ and $B^2 = B$, one quickly discovers that the term x_n (where $n \geq 2$) can be written as

$$??Ax_0 + ??BAx_0 + ??ABAx_0 + \cdots +$$

 MapleTech Vol. 4, No. 1, pp. 2–7; ISSN 1061-5733

$$??Ax_1+??BAx_1+??ABAx_1+\cdots+$$
$$??Bx_1+??ABx_1+??BABx_1+\cdots,$$

where the "??" stand for some nonnegative coefficients. Note each term in the first row ends in Ax_0, each term in the second row ends in Ax_1, and each term in the last row ends in Bx_1.

We collect all these coefficients in matrices named $A0$, $A1$, $B1$. We explain what the entries of these matrices are by an example. Consider

$$x_4 = \left(\frac{1}{4}\,Ax_0 + \frac{1}{8}\,(\,BA\,)\,x_0\right) + \left(\frac{1}{4}\,(\,BA\,)\,x_1\right) + \left(\frac{1}{8}\,Bx_1 + \frac{1}{4}\,(\,AB\,)\,x_1\right)$$

The *fourth* term of the sequence is x_4. It determines the *fourth* rows of the matrices. According to the coefficients, the *non-zero* entries in the fourth row of the matrices $A0$, $A1$, $B1$ are

$$A0[4,1]=\tfrac{1}{4},\ A0[4,2]=\tfrac{1}{8}, A1[4,2]=\tfrac{1}{4},$$
$$B1[4,1]=\tfrac{1}{8},\ B1[4,2]=\tfrac{1}{4}.$$

In other words, the (n,m)-entry $A0[n,m]$ of the matrix $A0$ (where $n \geq 2$) is the coefficient for x_n appearing in front of a product of m alternating operators applied to x_0 and ending in Ax_0:

$$x_n = \cdots + A0[n,m](\underbrace{\cdots ABA}_{m \text{ operators}})x_0 + \cdots.$$

We define the matrices $A1$ and $B1$ analogously. For convenience, we set the first row in every matrix identically equal to zero.

The determination of these coefficients — a trivial though tedious job — is easily achieved by the following Maple code:

```
> with(linalg): N := 40: # N matrix size.

Warning: new definition for   norm
Warning: new definition for   trace


> A0 := matrix(N,N,0): A1 := matrix(N,N,0):
> B1 := matrix(N,N,0):
> A0[2,1] := 1/2: B1[2,1] := 1/2:
> A0[3,2] := 1/4: A1[3,1] := 1/2:
>                 B1[3,1] := 1/4:
> for n from 4 to N do for m from 1 to N do
>   if type(m,odd) then # m is odd
>      if m=1 then
> A0[n,1] := (1/2)*A0[n-2,1];
> A1[n,1] := (1/2)*A1[n-2,1];
> B1[n,1] := (1/2)*B1[n-1,1];
>      else #m is odd and bigger than 1
> A0[n,m] := (1/2)*(A0[n-2,m]+A0[n-2,m-1]);
> A1[n,m] := (1/2)*(A1[n-2,m]+A1[n-2,m-1]);
> B1[n,m] := (1/2)*(B1[n-1,m]+B1[n-1,m-1]);
>      fi;
>   else # m is even
> A0[n,m] := (1/2)*(A0[n-1,m-1]+A0[n-1,m]);
> A1[n,m] := (1/2)*(A1[n-1,m-1]+A1[n-1,m]);
> B1[n,m] := (1/2)*(B1[n-2,m]+B1[n-2,m-1]);
>   fi;
> od; od;
```

Before we take a look at the so-created matrices, we test our code through the following "pretty output" routine:

```
> X := proc(n)
> global A0,A1,B1; local x,A,B,AB,BA,k,
> vA0,wA0,vA1,wA1,vB1,wB1;
> vA0 := row(A0,n); vA1 := row(A1,n);
> vB1 := row(B1,n); wA0 := vector(N);
> wA1 := vector(N); wB1 := vector(N);
> for k from 1 to N do
> if type(k,even) then
>   wA0[k] := ``(BA)^(k/2)*x[0];
>   wA1[k] := ``(BA)^(k/2)*x[1];
>   wB1[k] := ``(AB)^(k/2)*x[1];
> else
>   wA0[k] := ``(AB)^((k-1)/2)*Ax[0];
>   wA1[k] := ``(AB)^((k-1)/2)*Ax[1];
>   wB1[k] := ``(BA)^((k-1)/2)*Bx[1];
> fi;
> od;
> ``(innerprod(vA0,wA0)) +
> ``(innerprod(vA1,wA1)) +
> ``(innerprod(vB1,wB1));
> end:
```

This code works very well for exploring terms of the sequence. For instance, `> x[5]=X(5);` results in

$$x_5 = \left(\frac{3}{16}\,(\,BA\,)\,x_0 + \frac{1}{8}\,(\,AB\,)\,Ax_0\right) + \left(\frac{1}{4}\,Ax_1 + \frac{1}{8}\,(\,BA\,)\,x_1\right) + \left(\frac{1}{16}\,Bx_1 + \frac{1}{8}\,(\,AB\,)\,x_1 + \frac{1}{8}\,(\,BA\,)\,Bx_1\right)$$

The corresponding output for `>x[13]=X(13);` is already several lines long; the iterates become quickly very complex. Let us see part of the matrix $A0$ by issuing the command

```
> submatrix(A0,1..11,1..9);
```

$$\begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{1}{4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{1}{4} & \frac{1}{8} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{3}{16} & \frac{1}{8} & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{1}{8} & \frac{3}{32} & \frac{1}{16} & \frac{1}{16} & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{7}{64} & \frac{5}{32} & \frac{1}{16} & 0 & 0 & 0 & 0 & 0 \\
\frac{1}{16} & \frac{7}{128} & \frac{5}{64} & \frac{7}{64} & \frac{1}{32} & 0 & 0 & 0 & 0 \\
0 & \frac{15}{256} & \frac{17}{128} & \frac{3}{32} & \frac{1}{32} & \frac{1}{64} & 0 & 0 & 0 \\
\frac{1}{32} & \frac{15}{512} & \frac{17}{256} & \frac{29}{256} & \frac{9}{128} & \frac{3}{128} & 0 & 0 & 0 \\
0 & \frac{31}{1024} & \frac{49}{512} & \frac{23}{256} & \frac{1}{16} & \frac{3}{64} & \frac{1}{128} & 0 & 0
\end{bmatrix}$$

It is easy to spot the pattern in the first and second column of this matrix. However, already the third column is difficult to do "by eye". So let us extract the third column of the matrix $A0$ and multiply by powers of 2 to "integerize":

```
> l := col(A0,3):
> for n from 1 to N do l[n]:= l[n]*2^(n-2):
> od:
> eval(l);
```

[0 0 0 0 1 1 5 5 17 17 49 49 129 129 321 321 769 769 1793 1793 4097 4097 9217 9217 20481 20481 45057 45057 98305 98305 212993 212993 458753 458753 983041 983041 2097153 2097153 4456449 4456449]

Ignoring the double occurrences, the question arises: what is this sequence $1, 5, 17, 49, 129, 321, \ldots$?

Questions of this kind are often easy to answer provided one knows Sloane and Plouffe's *Encyclopedia of Integer Sequences* [11]: our sequence is labeled M3874 and its generating function is $1/((1-z)(1-2z)^2)$.

This clearly suggests trying to find generating functions for all columns of the matrix $A0$. (We recommend [6, 14] as excellent references on generating functions.) With the Maple package gfun [10], this is a matter of only a few lines:

```
> for m from 1 to 6 do
> gen_func_for_column.m := factor(
> guessgf( convert(col(A0,m),list), z)[1]);
> od;
```

$$\mathit{gen\_func\_for\_column1} := -\frac{z}{z^2-2}$$

$$\mathit{gen\_func\_for\_column2} := \frac{z^2}{(z-2)(z^2-2)}$$

$$\mathit{gen\_func\_for\_column3} := -\frac{z^4}{(z-2)(z^2-2)^2}$$

$$\mathit{gen\_func\_for\_column4} := \frac{z^5}{(z-2)^2(z^2-2)^2}$$

$$\mathit{gen\_func\_for\_column5} := -\frac{z^7}{(z-2)^2(z^2-2)^3}$$

$$\mathit{gen\_func\_for\_column6} := \frac{z^8}{(z-2)^3(z^2-2)^3}$$

Although this approach eventually fails, with the default series length parameter settings in gfun, already these few columns suggest the general form of the generating function. Using the common notation $[z^n]f(z) = f_n$ where $f(z) = \sum_n f_n z^n$, we can summarize our findings as follows:

Observation 2 Suppose $n \geq 1$. The entries of the matrix $A0$ are described by

$$A0[n, 2k+1] = [z^{n-1}]\left(\frac{-z^{3k+1}}{(z-2)^k(z^2-2)^{k+1}}\right), \ \forall k \geq 0;$$

and

$$A0[n, 2k] = [z^{n-1}]\left(\frac{z^{3k-1}}{(z-2)^k(z^2-2)^k}\right), \ \forall k \geq 1.$$

We were tempted to try to find closed forms for these column sequences. With Maple's genfunc package and its rsolve command, this is indeed possible in theory. We obtained closed forms for up to column 13 of the matrix $A0$; however, even the *simplified* closed forms go on for several pages. In other words, it is hopeless to expect "simple" closed forms. (The cause for our "despair" is explained in Remark 6.)

In contrast, rsolve does discover a closed form for the row sum of the matrix $A0$:

Observation 3 The n^{th} row sum of the matrix $A0$ is given by

$$\sum_m A0[n,m] = \frac{1}{3} + \frac{2}{3}\left(\frac{-1}{2}\right)^n, \quad \forall n \geq 2.$$

The procedure can be repeated for the other matrices, $A1$ and $B1$. Altogether, we obtain the following two results:

Theorem 4 The (n, m)-entries of the matrices $A0$, $A1$, $B1$ (where $n, m \geq 1$) are given by

	$m = 2k+1$
$A0[n,m]$	$[z^{n-1}]\left(\frac{-z^{3k+1}}{(z-2)^k(z^2-2)^{k+1}}\right)$
$A1[n,m]$	$[z^{n-1}]\left(\frac{-z^{3k+2}}{(z-2)^k(z^2-2)^{k+1}}\right)$
$B1[n,m]$	$[z^{n-1}]\left(\frac{-z^{3k+1}}{(z-2)^{k+1}(z^2-2)^k}\right)$

and

	$m = 2k$
$A0[n,m]$	$[z^{n-1}]\left(\frac{z^{3k-1}}{(z-2)^k(z^2-2)^k}\right)$
$A1[n,m]$	$[z^{n-1}]\left(\frac{z^{3k}}{(z-2)^k(z^2-2)^k}\right)$
$B1[n,m]$	$[z^{n-1}]\left(\frac{z^{3k}}{(z-2)^k(z^2-2)^k}\right)$

Theorem 5 The n^{th} row sums of the matrices $A0$, $A1$, $B1$ (where $n \geq 2$) are given by

matrix	n^{th} row sum
$A0$	$\frac{1}{3}+\frac{2}{3}\left(\frac{-1}{2}\right)^n$
$A1$	$\frac{1}{3}-\frac{4}{3}\left(\frac{-1}{2}\right)^n$
$B1$	$\frac{1}{3}+\frac{2}{3}\left(\frac{-1}{2}\right)^n$

Remark 6 We outline proofs of Theorem 4 and Theorem 5 rather than giving full details.

Let us start with Theorem 4. For each matrix, closed forms for the entries in the first columns are easily verified and so are the statements on the corresponding generating functions. This suggests a proof by induction on the column and already yields the base case. The induction step is done by considering two cases: the column is odd or it is even. (This can be avoided for the price of rather cumbersome formulae involving the floor function.) Having already a guess for the solution, the induction step is readily completed by arguing similarly to Graham et al.'s [6, Section 7.3, Example 3]. (Of course, using this approach also leads to the discovery of the generating functions!)

Theorem 5 is much easier to prove. Indeed, for each matrix, the corresponding recurrence relation for the row sum is of the form $r_n = \frac{1}{2}r_{n-1} + \frac{1}{2}r_{n-2}$ (with the appropriate boundary conditions). These recurrence relations are readily solved (either by hand or by Maple's `rsolve` command).

The results in [6, Section 7.3] — in particular, the General Expansion Theorem for Rational Generating Functions on page 341 — show in hindsight why it was naive to expect a general closed form for the entries in each matrix.

The main result

Fact 7 (Toeplitz, 1911) Suppose T is a *Toeplitz matrix*, i.e., an infinite matrix of real numbers $(t_{n,m})$ with

(i) $\lim_n t_{n,m} = 0, \quad \forall m;$

(ii) $\sup_n \sum_m |t_{n,m}| < +\infty;$

(iii) $\lim_n \sum_m t_{n,m} = r.$

Suppose further (y_n) is a sequence in X that converges to some $y \in X$. If the following series exist

$$z_n := \sum_m t_{n,m} y_m, \qquad \forall m,$$

then the sequence (z_n) converges to ry.

Remark 8 The classical Toeplitz result arises when $X = \mathbb{R}$ and $r = 1$; fortunately, the proof of [13, Theorem 7.85] works in the more general situation of Fact 7 just as well.

Remark 9 There is another kind of matrix commonly called a Toeplitz matrix, namely a matrix constant along diagonals. We do not use such matrices here.

We are now ready for the main result.

Theorem 10 The sequence (x_n) converges to

$$C(\tfrac{1}{3}x_0 + \tfrac{2}{3}x_1).$$

Proof. Recall that x_n (where $n \geq 2$) can be written as the sum of three sums in the following way:

$$\sum_m A0[n,m] \underbrace{\cdots ABA}_{m \text{ operators}} x_0 +$$
$$\sum_m A1[n,m] \underbrace{\cdots ABA}_{m \text{ operators}} x_1 +$$
$$\sum_m B1[n,m] \underbrace{\cdots BAB}_{m \text{ operators}} x_1.$$

On the one hand, Theorem 4 and Theorem 5 show that $A0$, $A1$, and $B1$ are Toeplitz matrices. On the other hand, Fact 1 implies that the corresponding three sequences of alternating projections $(x_0, Ax_0, BAx_0, \ldots)$, $(x_1, Ax_1, BAx_1, \ldots)$, $(x_1, Bx_1, ABx_1, \ldots)$ converge to Cx_0, Cx_1, Cx_1 respectively. Altogether, Fact 7 yields that (x_n) converges to $\frac{1}{3}Cx_0 + \frac{1}{3}Cx_1 + \frac{1}{3}Cx_1 = C(\frac{1}{3}x_0 + \frac{2}{3}x_1)$. □

Remark 11 If $X = U = V = \mathbb{R}$, then $A = B = I$ (here and later I stands for the *identity map*) and the iteration becomes $x_n = \frac{1}{2}x_{n-1} + \frac{1}{2}x_{n-2}$. In this case, we can directly deduce that (x_n) converges to $\frac{1}{3}x_0+\frac{2}{3}x_1$ (which is, of course, in accordance with Theorem 10 since $U \cap V = \mathbb{R}$ and hence $C = I$).

Remark 12 Theorem 10 shows that the most simple instance of an asynchronous projection method requires already a somewhat involved analysis. Results on more general asynchronous projection methods are thus likely very complicated. However, we believe that our approach will generalize to

- finitely many closed subspaces (instead of just 2);
- fixed weighted coefficients (instead of $\frac{1}{2}, \frac{1}{2}$); and
- idempotent operators (instead of orthogonal projections).

Computing orthogonal projections

In this section, we briefly describe how one actually computes orthogonal projections onto subspaces in $\mathbb{R}^n$.

Suppose a subspace U is given as the linear span of vectors (not necessarily linearly independent) which we collect as column vectors in a matrix M. Interpreting M as a linear mapping, we can identify U with the range of M. Then the orthogonal projection onto U is $MM^\dagger$, where $M^\dagger$ is the so-called *Moore/Penrose inverse of M*; see [7, Section II.2]. (Groetsch's monograph [7] is an excellent reference on Moore/Penrose inverses, which are also called *generalized* or *pseudo inverses.*)

There are various ways to compute the Moore/Penrose inverse. One particularly suitable for Maple is the *Tihonov Regularization*; see [9, Exercise 7.3.9] or [7, Example II.3.5]:

$$M^\dagger = \lim_{t\to 0^+} M^*(MM^* + tI)^{-1}.$$

The following Maple code describes the procedure `MPinv` which expects a (not too large) matrix and returns its Moore/Penrose inverse:

```
> MPinv := M -> map( limit,
>  evalm(transpose(M) &* (M &* transpose(M)
>                           + t*&*())^(-1)),
>        t=0, right):
```

(This is similar to the code in [8].)

As the reader might guess, computing orthogonal projections by hand is not much fun and prone to errors. There is another way of calculating the Moore/Penrose inverse using the *Singular Value Decomposition*; see [9, Exercise 7.3.7] or [5, Section 5.5.4]. However, implementations of this method are less useful for handling matrices with nonnumeric entries (but more suitable for large matrices). To illustrate the power of the Maple code, we examine a "symbolic example":

```
>M := matrix( [ [1,0] , [0,epsilon] ,
[0,0] ]);
```

$$M := \begin{bmatrix} 1 & 0 \\ 0 & \varepsilon \\ 0 & 0 \end{bmatrix}$$

The Moore/Penrose inverse of M is computed via

```
>MPinv(M);
```

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & \dfrac{1}{\varepsilon} & 0 \end{bmatrix}$$

This example demonstrates the great numerical sensitivity of Moore/Penrose inverses; see [5, Section 5.5.5] (the reader be warned that the formula for $M^\dagger$ given there contains a typo).

Conclusion

We have analyzed an asynchronous projection method iteration by following the steps below.

Step 1 To build intuition, we used Maple for computing iterates and generating data in form of (infinite-dimensional) matrices.

Step 2 We employed Maple's `gfun` package to study the obtained data; this allowed us to "guess" general generating function formulae for columns of the matrices. (The formulae can be proved rigorously, of course.)

Step 3 Using a result from Classical Analysis, we were able to prove that the iterates converge and to determine the limit.

The combinatorial work was thus entirely done by Maple. Also, the computation of the iterates is easy (via the Moore/Penrose inverses) because of Maple's symbolic capabilities.

In summary, Maple put an expert in generating functions on our desks suggesting a painless and fun analysis of the iteration that led to a complete solution.

Acknowledgment

We wish to thank Greg Fee, Luis Goddyn, Simon Plouffe, and two anonymous referees for helpful suggestions and comments. RMC thanks the CECM and the Department of Mathematics and Statistics at Simon Fraser University for sabbatical support.

References

[1] H.H. Bauschke and J.M. Borwein: On the convergence of von Neumann's alternating projection algorithm for two sets. *Set-Valued Analysis*, **1(2)**, pp. 185–212, (1993).

[2] H.H. Bauschke and J.M. Borwein: On projection algorithms for solving convex feasibility problems. *SIAM Review*, **38(3)**, pp. 367–426, September (1996).

[3] D. Chazan and W. Miranker: Chaotic relaxation. *Linear Algebra and its Applications*, **2(2)**, pp.199–222, April (1969).

[4] J.B. Conway: *A Course in Functional Analysis*, Volume 96 of *Graduate Texts in Mathematics*, second edition. Springer-Verlag, New York, (1990).

[5] G.H. Golub and C.F. Van Loan: *Matrix Computations*, volume 3 of *Johns Hopkins Series in the Mathematical Sciences*, second edition. The Johns Hopkins University Press, Baltimore and London, (1993).

[6] R.L. Graham, D.E. Knuth, and O. Patashnik: *Concrete Mathematics*, second edition. Addison-Wesley Publishing Company, (1994).

[7] C.W. Groetsch: *Generalized inverses of linear operators*. Monographs and textbooks in pure and applied mathematics, Vol **37**, Marcel Dekker, Inc., New York, (1977).

[8] A. Heck: *Introduction to Maple*. Springer-Verlag, New York, (1995).

[9] R.A. Horn and C.R. Johnson: *Matrix analysis*, second edition, Cambridge University Press, (1985).

[10] B. Salvy and P. Zimmermann: Gfun: a maple package for the manipulation of generating and holonomic functions in one variable. *ACM Transactions on Mathematical Software*, **20(2)**, pp. 163–177, 1994.

[11] N.J.A. Sloane and S. Plouffe: *The Encyclopedia of Integer Sequences*. Academic Press, Inc., (1995).

[12] J.C. Strikwerda: A convergence theorem for chaotic asynchronous relaxation. *Linear Algebra and its Applications*. To appear.

[13] K.R. Stromberg: *An introduction to classical real analysis*. Wadsworth International Group, Belmont, California, (1981).

[14] H.S. Wilf: *generatingfunctionology*. Academic Press, (1990).

Biographies

Heinz H. Bauschke received his Ph.D from Simon Fraser University (Department of Mathematics and Statistics) in 1996. He is currently an NSERC postdoc in the Department of Combinatorics & Optimization at the University of Waterloo.

His mathematical interests include convex analysis, algorithms for convex feasibility problems (in particular, projection methods), and symbolic computation.

Please see
`http://www.cecm.sfu.ca/~bauschke/`
for further information.

Professor Robert M. Corless got his Ph.D in classical applied mathematics from the Mechanical Engineering department of the University of British Columbia in 1987. Since then he has been a professor in the Applied Mathematics Department at the University of Western Ontario.

He is the author of more than sixty papers in the areas of flow-induced vibration, nonlinear dynamical systems, numerical analysis, symbolic computation, and the effective use of technology in mathematics education. His principal interests are now symbolic computation and the analysis of the reliability of numerical methods for the simulation of chaotic dynamical systems. His first book, *Essential Maple*, was published by Springer-Verlag in December 1994. More books are currently under way.

He is now Editor-in-Chief of the SIGSAM Bulletin (Communications in Computer Algebra), a quarterly publication of the ACM.

Please see
`http://pineapple.apmaths.uwo.ca/~rmc`
for more details.