Technische Universität Berlin
Institut für Mathematik

An Algebraic Multigrid Preconditioner
for a Class of Singular M-Matrices

Elena Virnik

Preprint 3-2006

An algebraic multigrid preconditioner for a class of singular M-Matrices

Elena Virnik

8th May 2006

Abstract

We apply algebraic multigrid (AMG) as a preconditioner for solving large singular linear systems of the type $(I-T^T)x = 0$ with GMRES. Here, T is assumed to be the transition matrix of a Markov process. Although AMG and GMRES are originally designed for the solution of regular systems, with adequate adaptation their applicability can be extended to problems as described above.

1 Introduction

This paper treats the application of an algebraic multigrid (AMG) method [11] as a preconditioner for solving large singular linear systems of the type

$$Ax := (I - T^T)x = 0 \tag{1}$$

with the generalised minimal residual (GMRES) method [12]. Here, T is the transition matrix of a Markov process. The applicability of GMRES to singular systems was examined, e.g., in [1, 4]. Our algebraic multigrid strategy is based on the classical theory in [11] originally designed for systems involving regular, symmetric, positive definite M-matrices. The approach in [11] was generalised to symmetric positive definite and semi-definite matrices without the M-matrix property, e.g., in [3, 5, 6, 10]. The main result of this paper is the extension of the AMG approach to a class of singular non-symmetric M-matrices.

The theory of Markov chains [2, 13] represents an extremely important tool that has a broad variety of applications not only in sciences, such as biology, physics or chemistry, but also in business and economics. As an example, we consider a Markov chain model for the blood circulation in a human body.

Example 1 *Let us imagine the organs within a human body as states of a blood particle. Within one time step, it can advance from one organ to another with a certain probability. It is also possible that it stays within the organ and only continues its journey a few time steps later. Hence, if we define a set of (sub)organs as states of the Markov chain and the corresponding transition probabilities, we get a large sparse (row) stochastic matrix. We are interested in the stationary distribution of the Markov chain, which would, e.g., represent the concentration of a dissolved substance after injection into an organ. For a more detailed description, see [15].*

1

2 Preliminaries

We call a vector $v \in \mathbb{R}^n$ *positive* and we write $v > 0$ if all entries v_i are positive. A matrix $T \in \mathbb{R}^{n \times n}$, $T = (t_{ij})_{i,j=1,\ldots,n}$ is called *positive (non-negative)* and we write $T > 0$ ($T \geq 0$) if all entries t_{ij} are positive (non-negative). A matrix $T \in \mathbb{R}^{n \times n}$ is called *reducible* if there exists a permutation matrix $P \in \mathbb{R}^{n \times n}$, such that $PTP^T = \begin{bmatrix} T_{11} & 0 \\ T_{21} & T_{22} \end{bmatrix}$. Otherwise it is called *irreducible*.

A scalar $\lambda \in \mathbb{R}$ is called *eigenvalue* of the matrix $T \in \mathbb{R}^{n \times n}$ if a vector $v \in \mathbb{R}^n$, $v \neq 0$ exists, such that $Tv = \lambda v$. The vector v is called *(right) eigenvector* of T associated with λ. Accordingly, a vector $w \in \mathbb{R}^n$, $w \neq 0$ with $w^T T = \lambda w^T$ is called *(left) eigenvector* of T. Let $T \in \mathbb{R}^{n \times n}$ have the eigenvalues λ_i, $i = 1, \ldots, n$. Then we call $\rho(T) = \max_{1 \leq i \leq n} |\lambda_i|$ the *spectral radius of T*.

A process is called *finite homogeneous Markov chain* if it has n states $s_1, \ldots, s_n$ and the transition probability $P[s_i \rightsquigarrow s_j] =: t_{ij}$ is time-independent. The matrix $T = [t_{ij}]_{i,j=1,\ldots,n}$ satisfies $t_{ij} \geq 0$ and $\sum_{j=1}^n t_{ij} = 1$ for $i, j = 1, \ldots, n$, i.e. is *(row) stochastic* and is called *transition matrix* of a Markov chain. We denote by $x^k = (x_i^k)$ the *probability distribution vector*, where x_i^k is the probability that the system is in state s_i after k steps. We have, $x_i^k \geq 0$ and $\sum_{i=1}^n x_i^k = 1$ for each k. A distribution vector x is said to be *stationary* if $x^T T = x^T$.

The well-known Perron-Frobenius Theorem guarantees the existence and uniqueness of a stationary distribution.

Theorem 1 (Perron-Frobenius Theorem [2], p. 27) *Let $T \geq 0$ be irreducible with spectral radius $\rho(T)$. Then $\rho(T)$ is a simple eigenvalue and T has a positive left and right eigenvector corresponding to $\rho(T)$.*

Corollary 1 ([2], p. 28) *A positive eigenvector x of a non-negative matrix T corresponds to $\rho(T)$.*

Corollary 2 *Every finite homogeneous Markov chain has a stationary probability distribution vector. If the transition matrix T of the process is also irreducible, then the stationary probability distribution vector is unique.*

We define by $\mathcal{Z}^{n \times n} = \{A = [a_{ij}] \in \mathbb{R}^{n \times n} : a_{ij} \leq 0, i \neq j\}$ the set of all real matrices with non-positive off-diagonal entries. Let $B \geq 0$ with spectral radius $\rho(B)$. A matrix A of the form $A = sI - B$, with $s > 0$, and $s \geq \rho(B)$ is called *M-Matrix*. If $s > \rho(B)$ then A is a *non-singular* M-Matrix and if $s = \rho(B)$ then A is a *singular* M-Matrix. Hence, in our case $A = (I - T^T)$ is a singular M-matrix.

3 AMG preconditioning

The GMRES method we use was introduced in [12]. A version with left preconditioning may look as follows:

Algorithm 1 (Preconditioned MGS-GMRES) *Input:* $A \in \mathbb{R}^{n \times n}$, $b \in \mathbb{R}^n$, *starting vector $x_0 \in \mathbb{R}^n$, preconditioner M*
Output: $x_m \in \mathbb{R}^n$ *solution approximate in the m-th step*

1. *Compute $r_0 = M^{-1}(b - Ax_0)$, $\beta := \|r_0\|_2$ and $v_1 := r_0/\beta$*
2. *Define the $(m + 1) \times m$ matrix $\bar{H}_m = \{h_{ij}\}_{1 \leq i \leq m+1, 1 \leq j \leq m}$. Set $\bar{H}_m = 0$.*
3. *For $j = 1, 2, \ldots, m$ Do:*
4.   *Compute $w_j := M^{-1}Av_j$*
5.   *For $i = 1, \ldots, j$ Do:*
6.   *$h_{ij} := (w_j, v_i)$*
7.   *$w_j := w_j - h_{ij}v_i$*
8.   *EndDo*
9.   *$h_{j+1,j} = \|w_j\|_2$. If $h_{j+1,j} = 0$ set $m := j$ and go to 12*
10.   *$v_{j+1} = w_j/h_{j+1,j}$*
11. *EndDo*
12. *Compute y_m the minimiser of $\|\beta e_1 - \bar{H}_m y\|_2$*

13. *Set $x_m = x_0 + V_m y_m$.*

As we will see in the following, the application of AMG in our case requires the property that the matrix A has row sums zero. Since $A = (I - T^T)$ has column sums zero instead, for the construction of the preconditioner we use the matrix A^T and then use M^T for the iteration process.

3.1 Basic framework of AMG

The algebraic multigrid (AMG) method that is presented here follows the concept introduced in [11] that is motivated by geometric multigrid, where a sequence of grids is constructed from the underlying geometry with corresponding transfer operators between the grids. The main idea of GMG is to remove the smooth error, that cannot be eliminated by relaxation on the fine grid, by coarse-grid correction. The solution process then as usual consists of pre-smoothing, transfer of residuals from fine to coarse grids, interpolation of corrections from coarse to fine levels, and optional post-smoothing. In contrast to geometric multigrid, the idea of AMG is to define an *artificial* sequence of systems of equations decreasing in size,

$$A^m u^m = b^m, \tag{2}$$

where the superscript m denotes the m-th iterate, directly from the underlying matrix. We have $A^1 = A$ and for $m = 1$ the system (2) is identical to (1). We call these equations "coarse grid" equations. The interpolation operator P and the restriction operator R define the transfer from finer to coarser grids and vice versa. More precisely, let $A^m \in \mathbb{R}^{n_m \times n_m}$, then

$$R^m : \mathbb{R}^{n_m} \to \mathbb{R}^{n_{m+1}}$$
$$P^m : \mathbb{R}^{n_{m+1}} \to \mathbb{R}^{n_m},$$

and the operator on the coarser grid is defined by

$$A^{m+1} = R^m A^m P^m \quad \in \mathbb{R}^{n_{m+1} \times n_{m+1}}. \tag{3}$$

Thus, we do not need a geometry behind the problem.

The AMG method consists of two main parts, the setup phase and the solution phase. During the setup phase, the coarse grids and the corresponding operators are defined. We describe this in Section 3.2. The solution phase

consists of a multilevel iteration. Here, as an example, we illustrate a two-level iteration, which involves the combination of a smoothing process, (e.g., Gauss-Seidel, see [7]) with a correction on a coarser grid:

$$x^{(k+1)} = x^{(k)} + B^{-1}(b^m - A^m x^{(k)}), \tag{4a}$$

$$x^{(k+2)} = x^{(k+1)} + P^m (A^{m+1})^{-1} R^m (b^m - A^m x^{(k+1)}). \tag{4b}$$

Here, the matrix $(A^{k+1})^{-1}$ is replaced by the recursive application of (4) to the solution of the coarse grid system. The number of recursive calls, which is the number of levels m, depends on the size and structure of the matrix. For our method, we use the V-cycle pattern (see, e.g., [8]): Other recursive patterns,
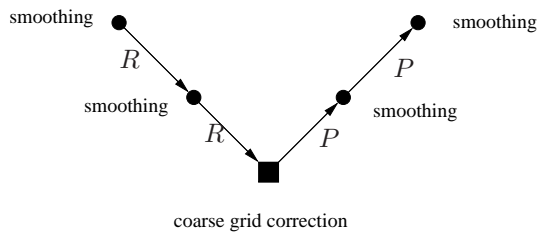
e.g., the W-cycle that involves two recursive calls per cycle are presented in [8].

3.2 Coarsening

The first main step of the coarsening process as introduced in [11] can be read as a permutation of the matrix $A \in \mathbb{R}^{n \times n}$ such that the fine grid nodes come first:

$$A \rightsquigarrow \pi^T A \pi = \left[\begin{array}{c|c} A_{FF} & A_{FC} \\ \hline A_{CF} & A_{CC} \end{array} \right],$$

We choose the fine grid nodes such that the block A_{FF} is close to a diagonal matrix. A further sparsification step is applied in which the remaining off-diagonal entries in A_{FF} are redistributed onto the A_{FC} block.

$$A \rightsquigarrow \tilde{A} = \left[\begin{array}{c|c} \tilde{A}_{FF} & \tilde{A}_{FC} \\ \hline A_{CF} & A_{CC} \end{array} \right]. \tag{5}$$

It is shown in [11] that the "sparsification" process leaves the row sums of the matrix A unchanged.

To obtain the restriction and interpolation operators, we consider the incomplete block LU decomposition of the matrix $\tilde{A}$:

$$\tilde{A} = \left[\begin{array}{c|c} \tilde{A}_{FF} & \tilde{A}_{FC} \\ \hline A_{CF} & A_{CC} \end{array} \right] = \left[\begin{array}{c|c} I & 0 \\ \hline A_{CF} \tilde{A}_{FF}^{-1} & I \end{array} \right] \left[\begin{array}{c|c} \tilde{A}_{FF} & \tilde{A}_{FC} \\ \hline 0 & S \end{array} \right] \tag{6}$$

where $S = A_{CC} - A_{CF} \tilde{A}_{FF}^{-1} \tilde{A}_{FC}$ is the Schur complement with respect to $\tilde{A}_{FF}$. We take S to be the new coarse grid operator. For other choices of restriction, interpolation and coarse grid operators see, e.g., [5, 6, 11, 14]. The following Lemma summarises the main results for the coarsening process.

Lemma 1 *Let A be sparsified and partitioned as in (6) and N be the number of coarse grid nodes. Let S as in (6) be chosen as the new coarse grid operator. Then, the following properties hold:*

1. *With the restriction and interpolation operators defined by*

$$R = \begin{bmatrix} -A_{CF}\tilde{A}_{FF}^{-1} & I \end{bmatrix} \in \mathbb{R}^{N \times n} \quad and \quad P = \begin{bmatrix} -\tilde{A}_{FF}^{-1}\tilde{A}_{FC} \\ I \end{bmatrix} \in \mathbb{R}^{n \times N},$$

the new coarse grid operator S can be calculated from

$$S = R\tilde{A}P \in \mathbb{R}^{N \times N}. \tag{7}$$

2. *The interpolation operator P has all row sums equal to one.*

3. *The new coarse grid operator S has row sums equal to zero.*

Proof.

1. By straight forward calculation we get

$$
\begin{aligned}
R\tilde{A}P &= \begin{bmatrix} -A_{CF}\tilde{A}_{FF}^{-1} & I \end{bmatrix} \begin{bmatrix} \tilde{A}_{FF} & \tilde{A}_{FC} \\ A_{CF} & A_{CC} \end{bmatrix} \begin{bmatrix} -\tilde{A}_{FF}^{-1}\tilde{A}_{FC} \\ I \end{bmatrix} = \\
&= \begin{bmatrix} -A_{CF}\tilde{A}_{FF}^{-1} & I \end{bmatrix} \begin{bmatrix} -\tilde{A}_{FC} + \tilde{A}_{FC} \\ -A_{CF}\tilde{A}_{FF}^{-1}\tilde{A}_{FC} + A_{CC} \end{bmatrix} \\
&= A_{CC} - A_{CF}\tilde{A}_{FF}^{-1}\tilde{A}_{FC} = S \in \mathbb{R}^{N \times N}.
\end{aligned}
$$

2. Let $\mathbb{1} = [1, \ldots, 1]^T$ represent the vector containing all entries equal to one. Hence, for all $i \in \{1, \ldots, N\}$ we have

$$0 = e_i^T \tilde{A}\mathbb{1} = \sum_{k \in \mathcal{C}_i} \tilde{a}_{ik} + \tilde{a}_{ii}$$

and thus, as all $\tilde{a}_{ik} \leq 0$ for $i \neq k$, we get

$$\frac{\sum_{k \in \mathcal{C}_i} |\tilde{a}_{ik}|}{\tilde{a}_{ii}} = 1,$$

where $\mathcal{C}_i$ is the set of interpolatory connections of i, i.e., we sum up the entries in the i-th row of the $\tilde{A}_{FC}$ block. Hence, all row sums of the interpolation operator P are equal to one.

3. From 2. we conclude that

$$e_i^T S\mathbb{1} = e_i^T R\tilde{A}P\mathbb{1} = e_i^T R\tilde{A}\mathbb{1} = 0 \quad \text{for all } i,$$

i.e. the zero row sums property is preserved for the new coarse grid operator S.

$\square$

3.3 Multilevel Setup

For the setup of a recursive multilevel cycle, it is essential to ensure that the important properties of the finest grid operator carry over to all coarser grids. In our case it is on the one hand the row sums zero property, which is important for an adequate interpolation of the smooth error and on the other hand the singular M-matrix property. In Lemma 1 we have shown that the coarsening process leaves the row sums zero. The following theorem states that the singular M-matrix property is preserved after the performance of one coarsening step.

Theorem 2 *Let $A = \begin{bmatrix} A_{FF} & A_{FC} \\ A_{CF} & A_{CC} \end{bmatrix}$ be a singular M-Matrix with zero row sums. Then $\tilde{A} = \begin{bmatrix} \tilde{A}_{FF} & \tilde{A}_{FC} \\ A_{CF} & A_{CC} \end{bmatrix}$ as in (5) and the corresponding new coarse grid operator $S = A_{CC} - A_{CF}\tilde{A}_{FF}^{-1}\tilde{A}_{FC}$ as in (7) are also singular M-matrices.*

Proof.

1. Show that $\tilde{A}$ is a singular M-Matrix:

   A is a singular M-matrix. Therefore, it can be expressed in the form

   $$A = sI - T, \quad \text{where} \quad T \text{ - transition matrix of a Markov process}$$
 $$\rho(T) = 1 = s.$$

   In [11] it was shown that the coarsening process leaves the row sums invariant, i.e. in our case equal to zero. Hence, we can write $\tilde{A} = I - \tilde{T}$, where $\tilde{T}$ has row sums one. Furthermore, we have $\tilde{A} \in \mathcal{Z}^{n \times n}$, i.e. the off-diagonal entries in $\tilde{A}$ are non-positive and the diagonal is non-negative (as otherwise the row sums could not be zero). The diagonal entries are also less than one. One can see this by considering the following "sparsification" steps (see [11] for a detailed description of the "sparsification" process):

   The initial matrix A has a sign and block structure as illustrated on the right. The sign "−" indicates that all entries in the corresponding section are non-positive. Accordingly, "+" signifies that the entries in the corresponding section are non-negative.
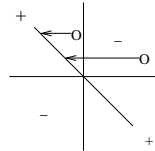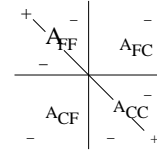
   In the first step, the weak neighbours in each row are added to the diagonal. The diagonal entries become smaller but stay non-negative.

   Then, off-diagonal entries of the A_{FF}-block are distributed onto the A_{FC}-block. The entries of the A_{FC}-block become smaller, but remain greater than -1, as otherwise the row sums could not be zero.
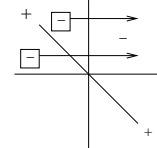
   Thus, $\tilde{T}$ is in addition non-negative and all entries are less than one. From this we conclude that $\tilde{T}$ is stochastic and thus, $\rho(\tilde{T}) = 1 = s$. Consequently, $\tilde{A}$ is a singular M-matrix.

2. Show that S as in (7) is a singular M-Matrix:

In Lemma 1 we have shown that $S\mathbb{1} = 0$, i.e. S has zero row sums. From this and from $A_{CC} \in \mathcal{Z}^{N \times N}$ and $A_{CF}\tilde{A}_{FF}^{-1}\tilde{A}_{FC} \geq 0$, we can conclude that $S = A_{CC} - A_{CF}\tilde{A}_{FF}^{-1}\tilde{A}_{FC} \in \mathcal{Z}^{N \times N}$. Let S now have the representation

$$S = sI - \hat{T}, \qquad \hat{T} \geq 0, \ \ s > 0.$$

Then, we get

$$S\mathbb{1} = (sI - \hat{T})\mathbb{1} = s\mathbb{1} - \hat{T}\mathbb{1} = 0$$
$$\Rightarrow \quad \hat{T}\mathbb{1} = s\mathbb{1}.$$

As $\mathbb{1} > 0$ is a positive eigenvector of $\hat{T}$, it follows from Corollary 1 that $s = \rho(\hat{T})$. Hence, S is a singular M-matrix. $\square$

From Theorem 2 we conclude that the singular M-matrix property is preserved on all coarser levels.

3.4 Integration of the Side Constraint into the Solution Process

For a badly chosen starting vector x_0 the GMRES method in Algorithm 1 may converge to the trivial solution, instead of the stationary distribution. In fact, for some problems we cannot choose the starting vector randomly but need to employ a specifically fixed starting vector as, e.g., in the case of the blood circulation model. Here, the initial distribution might represent an injection into a certain organ. Therefore, it makes sense to set $x_0 = [0, \ldots, 1, \ldots, 0]^T$, where the only nonzero entry $[x_0]_i = 1$ is positioned to represent the injection into the organ i. In this case the naive application of the preconditioned GMRES algorithm leads to a convergence to the trivial solution.

Hence, we have the side constraint $\mathbb{1}^T x = 1$ that we have to include into the solution process. The idea now is to shift the system such that the trivial solution is excluded as a possible solution. Consider the orthogonal projection matrix $Q := I - \frac{1}{n}\mathbb{1}\mathbb{1}^T$ with respect to $\mathbb{1}$, that is $\mathbb{1}^T Q = 0$. Any vector x can be decomposed into

$$x := Qy + z$$

such that $z = \alpha\mathbb{1}$ for some constant $\alpha \in \mathbb{R}$. In particular, for a vector $\tilde{x}$ that fulfils $\mathbb{1}^T\tilde{x} = 1$, we obtain $z := \frac{1}{n}\mathbb{1}$. Now we transform the initial problem as follows:

$$Ax = 0 \tag{8}$$
$$\Leftrightarrow \quad A(Qy + z) = 0$$
$$\Leftrightarrow \quad AQy = -\frac{1}{n}A\mathbb{1}.$$

By setting $\hat{A} := AQ$ and $b := -\frac{1}{n}A\mathbb{1}$, we get a formulation of the linear system where the side constraint is embedded:

$$\hat{A}y = b. \tag{9}$$

Now, we can solve this system instead of (8) using AMG preconditioned GM-RES. The system is still consistent since $b \in \text{im}(A)$. As Q is orthogonal with respect to z, we have $\tilde{x} = Qy + z \neq 0$ for all y.

Note, that in the implementation we do not compute the product $\hat{A} := AQ$ explicitly as Q is not sparse and matrix-matrix products are expensive, but rather successively calculate the corresponding matrix-vector products.

3.5 Direct Solution on the Coarsest Level

On the coarsest level we solve the singular system

$$A^m x^m = b^m \tag{10}$$

directly using the LU decomposition. From Section 3.3 we know that the operator A^m on the coarsest level is still a singular M-matrix. In theory, the coarsening process does not necessarily guarantee irreducibility of A^m, although in our practical examples this is the case. Since the treatment of this problem is beyond the scope of this paper, in the following discussion we will assume that A^m is irreducible. Then, an LU decomposition of A^m with L non-singular exists [9] and looks as follows:

$$A^m = LU = \begin{bmatrix} 1 & & \\ & \diagdown & \\ & & 1 \end{bmatrix} \begin{bmatrix} \diagdown & \\ & \diagdown \\ & & 0 \end{bmatrix}$$

To solve the system (10), we only need to prove that it is consistent. That is, we need to ensure that L^{-1} applied to the right hand side produces a zero in the last entry of the resulting vector.

Lemma 2 *Let A be a singular M-matrix with column sums zero and let $A = LU$ be an LU decomposition of A with L non-singular. Then*

$$e_n^T L^{-1} = \mathbb{1}^T.$$

Proof. $A = LU \Leftrightarrow L^{-1}A = U$. If we apply the n-th unit vector from the left, we get

$$e_n^T L^{-1} A = e_n^T U = [0, \ldots, 0].$$

Thus, it must hold that $e_n^T L^{-1} = \alpha \mathbb{1}^T$ for some scalar α. Yet, as L^{-1} has ones on the diagonal, we get $\alpha = 1$ and hence $e_n^T L^{-1} = \mathbb{1}^T$. $\square$

By construction, the AMG ensures that on any level j, the system (10) satisfies $\mathbb{1}^T b^j = 0$ since this holds for the initial right hand side in (9) and the new right hand sides on coarser grids are obtained via restriction of the residual, i.e.,

$$b^{j+1} = R^j (b^j - A^j x^k).$$

Note, that Lemma 1 is applied to A^T. Thus, R^j corresponds to $(P^j)^T$ in Lemma 1.

3.6 Numerical tests

For numerical tests, we used MATLAB® Version 6.5 run on a PC with an AMD Athlon(TM) XP 2100+ processor with relative machine precision eps $= 2.2204 \times 10^{-16}$. As convergence criterion for GMRES we used the tolerance $t = \|r_0\|\sqrt{\texttt{eps}}$, where r_0 is the initial residual, i.e. GMRES has converged when the residual norm falls under the tolerance t. The time measurement was conducted via the commands `clock` and `etime` and represents an upper bound for the required time. It can be further reduced by a more efficient implementation of the AMG algorithm. In Section 1, we have discussed the Markov chain model of the blood circulation in a human body. Table 1 compares some results of the basic GMRES solver with the AMG preconditioned GMRES applied to transition matrices of Markov chains that arise from the blood circulation problem.

Table 1: Blood-flow

		GMRES		AMG+GMRES				$\sum$AMG
n	nnz	TIME	ITS	GRID	$\#level$	TIME	ITS	TIME
88	182	0.07	28	0.29	4	0.02	2	0.31
122	250	0.07	37	0.35	4	0.02	2	0.37
314	634	0.23	101	0.67	5	0.03	2	0.7
614	1234	0.64	169	1.3	6	0.06	2	1.36
1125	2256	**5.15**	368	2.12	7	0.15	2	**2.27**
2598	5202	**34.41**	687	4.58	8	0.64	2	**5.22**
4952	9901	**119.79**	1000	10.6	10	5.45	2	**16.05**

In the first column of Table 1 we have the size n of the problem. The second column states the number nnz of nonzero elements in the matrix. The third and fourth columns contain the computation time and the number of iterations for the basic GMRES. The column labeled GRID shows the time required by AMG for the grid construction. In $\#level$ the number of constructed grids is given. The following TIME and ITS columns represent the computation time of AMG preconditioned GMRES and the corresponding number of iterations. The last column summarises the total time required by AMG preconditioned GMRES.

As we can see, the number of iterations required by basic GMRES grows linearly with the size of the matrix, whereas the number of iterations for the preconditioned GMRES remains constant. Yet, with respect to the overall computation time, the preconditioning becomes efficient only for large matrices. This is due to the fact, that the overall computation time of the AMG preconditioned GMRES is dominated by the time required to construct the "algebraic grid". The computational effort required for grid construction, however, although it also grows linearly, grows much slower with the matrix size than the computational effort for GMRES without preconditioning.

4 Conclusions

In this paper, the application of the AMG method as a preconditioner for solving large singular linear systems of the type $(I - T^T)x = 0$ with GMRES was examined. In doing so, we concentrated on the case, where T is the transition matrix of a Markov process. The AMG method [11] was originally developed for regular, symmetric, positive definite M-matrices. In our case, although the matrix $(I - T^T)$ is singular and non-symmetric, it is a singular M-matrix that possesses a number of useful properties. It turns out, that with the adaptation that is discussed in Sections 3.3-3.5, the method becomes applicable to singular matrices such as those in the blood circulation model. Numerical experiments illustrate that preconditioning with AMG method leads to significant acceleration of the convergence speed.

Acknowledgements

References

[1] V. A. Barker. Numerical solution of sparse singular systems of equations arising from ergodic Markov chains. *Comm. Statist. Stochastic Models*, 5(3):335–381, 1989.

[2] A. Berman and R. J. Plemmons. *Nonnegative Matrices in the Mathematical Sciences.* Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1994. (Revised reprint of the 1979 original).

[3] M. Brezina, R. Falgout, S. MacLachlan, T. Manteuffel, S. McCormick, and J. Ruge. Adaptive algebraic multigrid. *SIAM J. Sci. Comput.*, 27(4):1261–1286 (electronic), 2006.

[4] P. N. Brown and H. F. Walker. GMRES on (nearly) singular systems. *SIAM J. Matrix Anal. Appl.*, 18(1):37–51, 1997.

[5] Q. Chang and Z. Huang. Efficient algebraic multigrid algorithms and their convergence. *SIAM J. Sci. Comput.*, 24(2):597–618 (electronic), 2002.

[6] Q. Chang, Y. S. Wong, and H. Fu. On the algebraic multigrid method. *J. Comput. Phys.*, 125(2):279–292, 1996.

[7] G. H. Golub and C. F. Van Loan. *Matrix Computations.* Johns Hopkins University Press, Baltimore, MD, 1983.

[8] W. Hackbusch. *Multigrid Methods and Applications.* Springer-Verlag, Berlin, 1985.

[9] I Wen Kuo. A note on factorizations of singular M-matrices. *Linear Algebra and Appl.*, 16(3):217–220, 1977.

[10] S. MacLachlan, T. Manteuffel, and S. McCormick. Adaptive reduction-based AMG. *Num. Lin. Alg. Appl.*, to appear.

[11] J. W. Ruge and K. Stüben. Algebraic multigrid. In S. F. McCormick, editor, *Multigrid methods*, volume 3 of *Frontiers Appl. Math.*, pages 73–130. SIAM, Philadelphia, PA, 1987.

[12] Y. Saad and M.H. Schultz. GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing*, 7:856–869, 1986.

[13] W. J. Stewart. *Introduction to the Numerical Solution of Markov Chains*. Princeton University Press, Princeton, NJ, 1994.

[14] U. Trottenberg, C. W. Oosterlee, and A. Schüller. *Multigrid*. Academic Press Inc., San Diego, CA, 2001.

[15] E. Virnik. Algebraic multigrid preconditioners for computing stationary distributions of Markov processes. Diplomarbeit, Technische Universität Berlin, 2004. http://www.math.tu-berlin.de/preprints/abstracts/Report-10-2005.rdf.html.