TECHNISCHE
UNIVERSITÄT
MÜNCHEN

INSTITUT FÜR INFORMATIK

Sonderforschungsbereich 342:
Methoden und Werkzeuge für die Nutzung
paralleler Rechnerarchitekturen

Approximate Sparsity Patterns for the Inverse of a Matrix and Preconditioning

Thomas Huckle

TUM-I9829
SFB-Bericht Nr. 342/12/98 A
November 98

TUM–INFO–11-I9829-150/1.–FI

Anforderungen an: Prof. Dr. A. Bode
Sprecher SFB 342
Institut für Informatik
Technische Universität München
D-80290 München, Germany

Druck: Fakultät für Informatik der
Technischen Universität München

Approximate Sparsity Patterns for the Inverse of a Matrix and Preconditioning

Thomas Huckle
TU München, Institut für Informatik
Arcisstr. 21, D-80290 München, Germany
e-mail: huckle@informatik.tu-muenchen.de

Keywords: sparse approximate inverse, preconditioning

ABSTRACT

We consider a general sparse matrix A. Computing a sparse approximate inverse matrix M by minimizing $||AM - E||$ in the Frobenius norm is very useful for deriving preconditioners in iterative solvers, especially in a parallel environment. The problems, that appear in this connection in a distributed memory setting, are the distribution of the data - mainly submatrices of A - on the different processors. An a-priori knowledge of the data that has to be sent to a processor would be very helpful in this connection in order to reduce the communication time.

In this paper, we compare different strategies for choosing a-priori an approximate sparsity structure of A^{-1}. Such a sparsity pattern can be used as a maximum pattern of allowed entries for the sparse approximate inverse M. Then, with this maximum pattern we are able to distribute the claimed data in one step to each processor. Therefore, communication is necessary only at the beginning and at the end of a subprocess.

Using the characteristic polynomials and the Neumann series associated with A and $A^T A$, we develop heuristic methods to find good and sparse approximate patterns for A^{-1}. Furthermore we exactly determine the submatrices that are used in the SPAI algorithm to compute one new column of the sparse approximate inverse. Hence, it is possible to predict in advance an upper bound for the data that each processor will need.

Based on numerical examples we compare the different methods with regard to the quality of the resulting approximation M.

1. SPARSE APPROXIMATE INVERSES AND LINEAR EQUATIONS

We consider the problem of solving a system of linear equations $Ax = b$ in a parallel environment. Here, the $n \times n$-matrix A is large, sparse, unstructured, nonsymmetric, and ill-conditioned. The solution method should be robust, easy to parallelize, and applicable as a black box solver.

Direct solution methods like the Gaussian Elimination are often not very effective in a parallel environment. This is caused by the sequential nature of the computation and solution of a triangular factorization $A = LR$ with lower and upper triangular matrices L and R. Therefore, iterative solution methods like GMRES, BiCGSTAB, or QMR (see [3]) are frequently used..

For many important iterative methods the convergence depends heavily on the location of the eigenvalues of A. Therefore, the original system $Ax = b$ is replaced by an equivalent system $MAx = Mb$ or the system $AMz = b$, $x = Mz$. Here, the matrix M is called a preconditioner and has to satisfy three conditions:
- AM (or MA) should have a 'clustered' spectrum,
- M should be efficiently computable in parallel,
- "$M\times$Vektor" should be fast to compute in parallel.

Often used preconditioners are block-Jacobi-peconditioners, polynomial-preconditioners, or incomplete LU-decompositions of A [3]. But these preconditioners either lead to unsatisfactory convergence or are hard to parallelize.

A very promising approach is the choice of sparse approximate inverses for preconditioning, $M \approx A^{-1}$ and M sparse [6, 5, 9, 8, 10, 1, 4]. Then, in the basic iterative scheme only matrix-vector multiplications with M appear and it is not necessary to solve a linear system in M like in the incomplete LU-approach. Obviously, A^{-1} is a full matrix in general, and hence not for every sparse matrix A there will exist a good sparse approximate inverse matrix M. But the following scheme for computing a sparse approximate inverse, the so called SPAI algorithm, provides also information about the quality of the determined approximation [9].

We can compute such a matrix M by solving a minimization problem of the form $min\|AM - E\|$ for a given sparsity pattern for M. By choosing the Frobenius norm we arrive at an analytical problem that is very easy to solve. Furthermore, in view of

$$\min \|AM - E\|_F^2 = \sum_{k=1}^{n} \min \|AM_k - e_k\|^2$$

this minimization problem can be solved columnwise for M_k and is therefore embarrassingly parallel.

First, we consider M with a prescribed sparsity pattern, e.g. $M = 0$, M a diagonal matrix, or M with the same sparsity pattern as A or A^T. We get the columnwise minimization problems $\|AM_k - e_k\|_2$, $k = 1, 2, ..., n$, with a prescribed sparsity pattern for the column vector M_k. Let us denote by J_k the small index set of allowed nonzero entries in M_k, and the "reduced" vector of the nonzero entries by $\hat{M}_k := M_k(J_k)$. The corresponding submatrix of A is $A(:, J_k)$, and most of the rows of $A(:, J_k)$ will be zero in view of the sparsity of A. Let us denote the row indices of nonzero rows of $A(:, J_k)$ by I_k, and the corresponding submatrix by $\hat{A} = A(I_k, J_k)$, and the corresponding reduced vector by $\hat{e}_k = e_k(I_k)$. Hence, for the k-th column of M we have to solve the small least squares problem

$$\min \|\hat{A}\hat{M}_k - \hat{e}_k\| .$$

For the general case it is not possible to prescribe a promising sparsity pattern without causing J_k and I_k to be very large. This would result in large LS-problems and a very expensive algorithm. Therefore, for a given index set J_k with optimal solution $M_k(J_k)$ we need a dynamic procedure to find new promising indices that should be added to J_k. Then, we update I_k and solve the enlarged LS-problem until the residual $r_k = AM_k - e_k$ is small enough or until J_k gets too large. In general the start sparsity pattern should be $J_k = \emptyset$. Only for matrices with nonzero diagonal entries we can set $J_k = \{k\}$ in the beginning.

We use a hierarchy of three different criteria for finding new promising indices to be added to J_k. As a global a-priori criterion for M we only allow indices that appear in a given pattern S. Such global criteria are very helpful for distributing the data to the corresponding processors in a parallel environment because the parallel implementation is highly nontrivial in general [2, 4]. For the maximum allowed index set J_{max} we get a row index set I_{max} and a submatrix $A(I_{max}, J_{max})$, which represents the part of A that is necessary for the corresponding processor to compute M_k. If one processor has to compute M_k for several $k \in K$, then this processor only needs the submatrix of A that is given by the column indices $\cup_{k\in K} J_{max}$ and the row indices $\cup_{k\in K} I_{max}$. In the main part of this paper we will present and compare different strategies for choosing such a pattern S.

Now let us assume that we have already computed an optimal solution M_k with residual r_k of the LS-problem relative to an index set J_k. As a second local a priori criterion we consider only indices j with $(r_k^T A e_j)^2 > 0$. We will see later that this condition guarantees that the new index set $J_k \cup \{j\}$ leads to a smaller residual r_k.

The final selection of new indices out of the remaining index set, after applying the a-priori criteria, is ruled by

(a) a 1-dimensional minimization $\min_{\mu_j} \|A(M_k + \mu_j e_j) - e_k\|$, [6, 9, 10] or

(b) the full minimization problem $\min_{J_k\cup\{j\}} \|A\tilde{M}_k - e_k\|$ [8].

In the case (a) we consider

$$\min_{\mu_j \in \mathbf{R}} \|r_k + \mu_j A e_j\|_2 = \min_{\mu_j} \|A(M_k + \mu_j e_j) - e_k\|_2 =: \rho_j \ .$$

For every j the solution is given by

$$\mu_j = -\frac{r_k^T A e_j}{\|Ae_j\|_2^2} \quad \text{and} \quad \rho_j^2 = \|r_k\|_2^2 - \Big(\frac{r_k^T A e_j}{\|Ae_j\|_2}\Big)^2 \ .$$

Hence, indices with $(r_k^T A e_j)^2 = 0$ lead to no improvement in the 1-D minimization. We arrange the new possible indices j relative to the size of their corresponding residuals ρ_j. Similarly, the case (b) can be analysed [8, 10].

Now we have a sorted list of possible new indices. Starting with the smallest ρ_j we can add one or more new indices to J_k and solve the enlarged LS-Problem. Numerical examples show that it saves operations if more new indices are added per step.

2. A-PRIORI PATTERNS FOR SPARSE APPROXIMATE INVERSES

We are interested in finding small sparsity patterns that allow a good approximation of A^{-1}. There are well-known theoretical results on the exact relation between the pattern of A^{-1}, A and A^T [7]. But here we do not need the exact pattern of A^{-1} but a sparse pattern that allows a good approximation on A^{-1}. To make thinks easy we assume that the nonzero elements of A are greater than zero. Under the assumption $A \geq 0$ the following results are true, and for general A the same results will nearly be fulfilled by the effect of rounding errors.

First let us consider the characteristic polynomial for A. This leads to coefficients α_j that satisfy

$$0 = \alpha_n A^n + ... + \alpha_1 A + \alpha_0 E \quad \text{and} \quad A^{-1} = -(\alpha_n A^{n-1} + ... + \alpha_1 E)/\alpha_0 \ .$$

Therefore the pattern of A^{-1}, denoted by $S(A^{-1})$, is contained in the pattern $\bigcup_{j=0}^{n-1} S(A^j)$, or

$$S(A^{-1}) \subseteq S((E + A)^{n-1})$$

In view of the monotonic increasing sequence

$$S(E) \subseteq S((E + A)) \subseteq ... \subseteq S((E + A)^{n-1})$$

we can choose $S((E + A)^m)$ for a small m as an approximate pattern for A^{-1}

Similarly the Neumann representation

$$A^{-1} = \alpha \sum_{j=0}^{\infty} (E - \alpha A)^j$$

for small α shows that numerically $S((E + A)^j)$ are nearly contained in $S(A^{-1})$ for all j, and $S((E + A)^{n-1})$ is nearly equal to $S(A^{-1})$.

In the same way we can use the characteristic polynomial and the Neumann series for $B = A^T A$. This shows that

$$0 = \beta_n B^n + ... + \beta_1 B + \beta_0 E \quad \text{and} \quad B^{-1} = -(\beta_n B^{n-1} + ... + \beta_1 E)/\beta_0$$

which yields

$$A^{-1} = (\beta_n (A^T A)^{n-1} A^T + ... + \beta_1 A^T)/\beta_0 \ .$$

Furthermore in view of $A^{-1} = (A^T A)^{-m-1} (A^T A)^m A^T$, the pattern of $(A^T A)^m A^T$ is nearly contained in $S(A^{-1})$. Thus we have

$$S(A^{-1}) \subseteq S((A^T A)^{n-1} A^T)$$

and we get the monotonic increasing sequence

$$S(A^T) \subseteq S((A^T A) A^T) \subseteq ... \subseteq S((A^T A)^{n-1} A^T) \ .$$

Hence we can choose $S((A^TA)^mA^T)$ for a small m as approximate pattern for A^{-1}.

Now for orthogonal A the pattern of A^{-1} is equal to the pattern of A^T which can be totally different from $S((E+A)^j)$ for all $j < n-1$. Consider for example the $n \times n$ permutation matrix

$$A = \begin{pmatrix} 0 & 1 \\ E_{n-1} & 0 \end{pmatrix} .$$

Then $S((E+A)^j) \cap S(A^{-1}) = \emptyset$ for $j < n-1$, but $A^{n-1} = A^{-1}$. Therefore, it seems to be advisable to include in many cases the pattern of A^T in an a-priori guess for the pattern of A^{-1}. For $S = S(E+A)$ this can be done by replacing $E+A$ by $E+A+A^T$. Then we have the relations

$$S((A^TA)^mA^T) \subseteq S((E+A+A^T)^{2m})$$

and if the diagonal elements of A are all nonzero then also

$$S((E+A+A^T)^m) \subseteq S((A^TA)^m) .$$

Again for α small enough it holds

$$A^{-1}(A^{-T}A^{-1} - \alpha(A^{-1}+A^{-T}))^{-1}A^{-T} = (E - \alpha(A+A^T))^{-1} = \alpha \sum_{j=0}^{\infty} \alpha^j (A+A^T)^j .$$

This leads to the relation

$$A^{-1} = \alpha \sum_{j=0}^{\infty} \alpha^j ((A+A^T)^j A^T) \; (A^{-T}A^{-1} - \alpha(A^{-1}+A^{-T})) .$$

Therefore the approximations $S((E+A+A^T)^mA^T)$ and $S(A^T(E+A+A^T)^m)$ can be used for A^{-1} which are closely related to $S((E+A+A^T)^m)$.

For matrices with symmetric sparsity pattern and nonzero-diagonal entries these methods lead to the same approximate patterns. If the pattern is symmetric but there appear zero diagonal entries then the patterns differ slightly.

If the pattern itself is nonsymmetric then we can define a further approximate pattern in the following way. Assume that $A - A^T$ is regular. Then

$$A^{-1} = ((A-A^T)A)^{-1}(A-A^T) = \alpha \sum_{j=0}^{\infty} (E - \alpha(A-A^T)A)^j (A-A^T)$$

which leads to

$$S((E + |A-A^T|A)^m |A-A^T| = S((E+\Delta A)^m \Delta) .$$

For triangular matrices the inverse A^{-1} will be triangular, too. Hence, we should not use A^T for the pattern. As a-priori guess we can instead consider $|A|^m$ for a small m.

For computing the sparsity patterns described above we consider the graphs related to these matrices. Especially we can use the undirected graphs connected with the symmetric matrices $A^T A$ and $E + |A| + |A^T|$ (which is nothing else then the reflexive undirected graph connected with A).

3. PATTERN APPROXIMATION BY SPARSE APPROXIMATE INVERSES

If we follow the computations in the SPAI algorithm we can immediately read off an upper bound for the possible pattern of M. Let us denote by $M^{(0)}$ the start solution with pattern $S(M^{(0)})$. For every column M_k we allow only new entries at positions with $0 \neq e_j^T A^T r_k = e_j^T A^T (AM_k^{(0)} - e_k)$. Hence, if we would add all these new entries the new enlarged pattern would be $S(A^T AM_k^{(0)} - A^T e_k)$ for the k-th column and thus $S(M^{(1)}) \subset (S(A^T AM^{(0)}) \cup S(A^T))$. If in a second step we add new entries to $M^{(1)}$ we get $S(M^{(2)}) \subset (S(A^T AM^{(1)}) \cup S(A^T))$, and in general

$$S(M^{(\mu)}) \subset S((A^T A)^\mu M^{(0)}) \cup S((A^T A)^{(\mu-1)} A^T) \ .$$

This gives an upper bound for the sparsity pattern that can occur in the SPAI algorithm with μ steps of adding new entries in M.

For special start sparsity we can further analyse the resulting pattern:
$M^{(0)} = 0 : \quad S(M^{(\mu)}) \subset S((A^T A)^{\mu-1} A^T)$; this coincides with the result of the previous section.
$S(M^{(0)}) = S(A^T) : \ S(M^{(\mu)}) \subset S((A^T A)^\mu A^T)$.
$S(M^{(0)}) = S(diag(A)) = S(E) : \ S(M^{(\mu)}) \subset (S(A^T A)^\mu) \cup S((A^T A)^{\mu-1} A^T))$.

Hence, for every start sparsity pattern $M^{(0)}$ and for the maximum number μ of sweeps where we add new entries, we have an upper bound for the pattern that can occur in $M^{(\mu)}$. This upper bound is basically given by the pattern of $(A^T A)^{\mu-1} A^T$.

On the one hand we can use this information for providing every processor with that submatrix of A that is necessary to compute his columns M_k; on the other hand this is a purely algebraic method to find sparse patterns for approximating A^{-1}.

We can extend these results also to the symmetric positive definite case. Here, we want to determine approximate Cholesky factors L for A^{-1}. We replace the Frobenius norm minimization by the minimization of the Kaporin functional [12, 1]

$$\min \frac{(1/n)\, trace(L^T AL)}{det(L^T AL)^{(1/n)}} \ .$$

Again we can try to extend a given pattern for the lower triangular matrix L which leads to the condition $e_j^T ALe_k \neq 0$ [11]. If we begin with a start solution $L^{(0)}$ then again the maximum pattern in the next step is given by $S(L^{(1)}) \subset S(AL^{(0)})$. Let us denote by $low(A)$ the lower triangular $n \times n$ part of A and by $diag(A)$ the diagonal part of A. Then we can conclude that

$$S(L^{(1)}) \subset S(low(AL^{(0)})) \subset S(AL^{(0)})$$

or in general

$$S(L^{(\mu)}) \subset S(low(A\; low(A \ldots low(A\; low(AL^{(0)}))))) \;.$$

For start sparsity $S(E)$ this gives

$$S(L^{(\mu)}) \subset S(low(A\; low(A \ldots low(A\; low(A))))) \;.$$

4. NUMERICAL EXAMPLES

First let us compare the matrices $(A^T A)^m A^T$ with A^{-1} and with an approximate inverse M we get by the SPAI algorithm. We use the patterns $S((A^T A)^m A^T)$ as an upper bound S and apply the SPAI-algorithm for approximating A^{-1} restricted to this index set S. We try to compute M_k such that $\|AM_k - e_k\| \leq \epsilon$; we allow only nonzero entries at positions that occur in S, and as long as $nnz(M_k) \leq to$. As matrix we use ORRSIRR2 from the Harwell-Boeing collection.

The first pattern is the pattern of the larger entries of A^{-1} where we drop all entries smaller than 0.002, resp. 0.0015. The next figures show the full pattern $S(A^T)$ and $S(A^T AA^T)$, and the pattern of the larger entries in $S(A^T AA^T)$ and $S((A^T A)^2 A^T)$. The last figures show the patterns of the sparse approximate matrix M computed with the SPAI algorithm and parameters $\epsilon = 0.2$ and $to = 10$ where we use different patterns as upper bounds for the pattern of M.

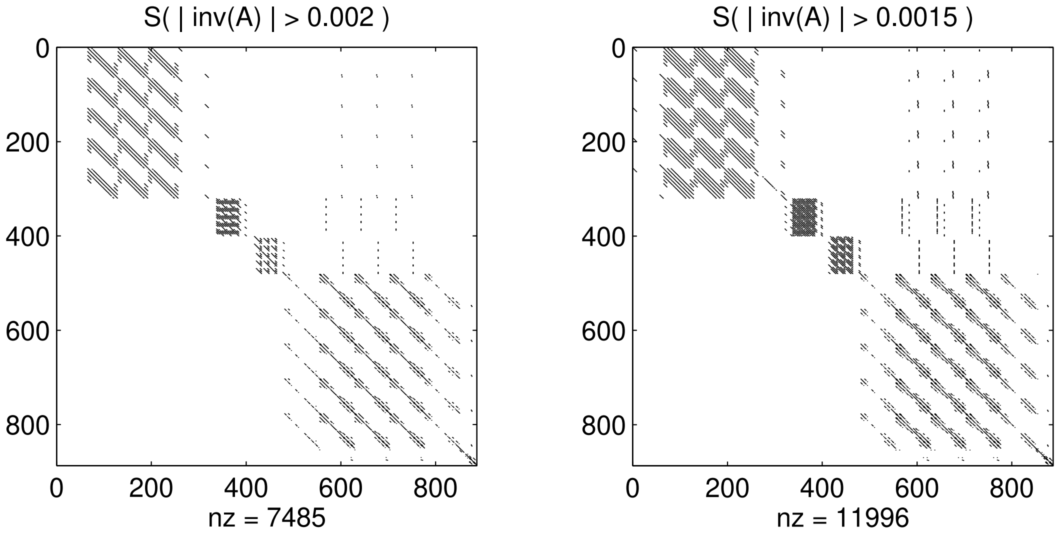

Figure 1.: Sparsity pattern $S(|A^{-1}| > 0.002$ and $S(|A^{-1}| > 0.0015$.

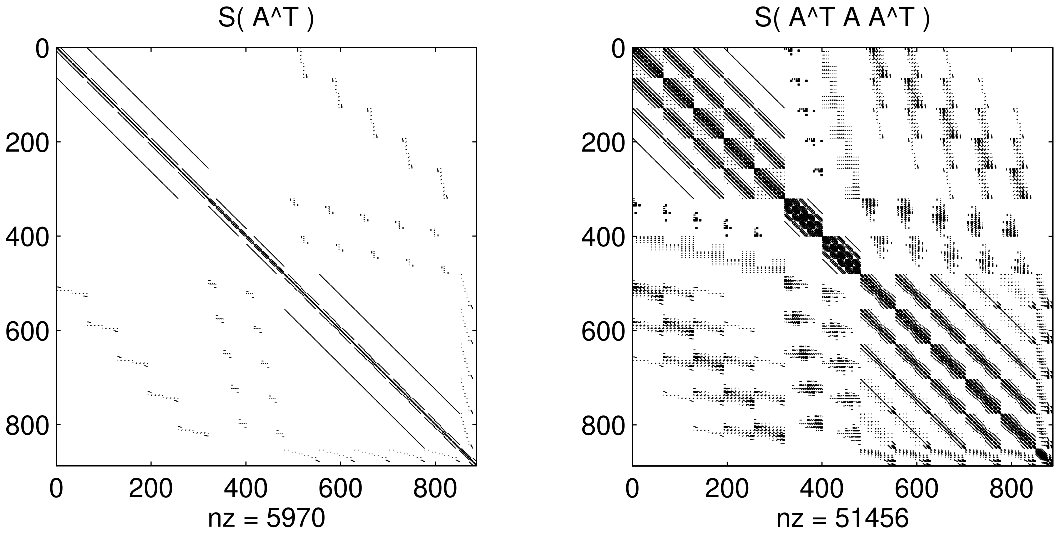

Figure 2.: Pattern $S(A^T)$ and $S(A^T A A^T)$.

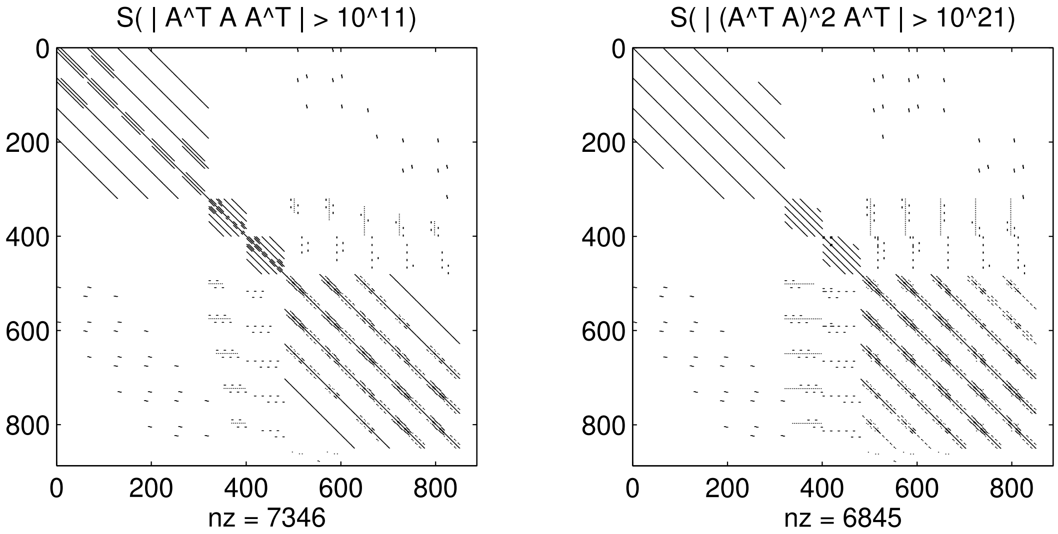

Figure 3.: Pattern $S(|A^T A A^T| > 10^{11})$ and $S(|(A^T A)^2 A^T| > 10^{21})$.

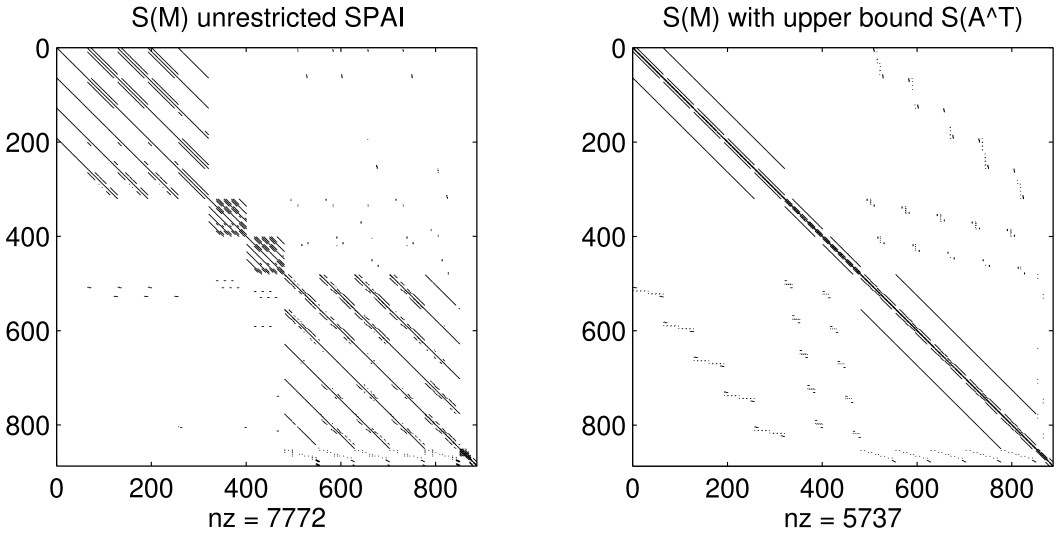

Figure 4.: $S(M)$ for unrestricted SPAI and with upper bound $S(A^T)$.

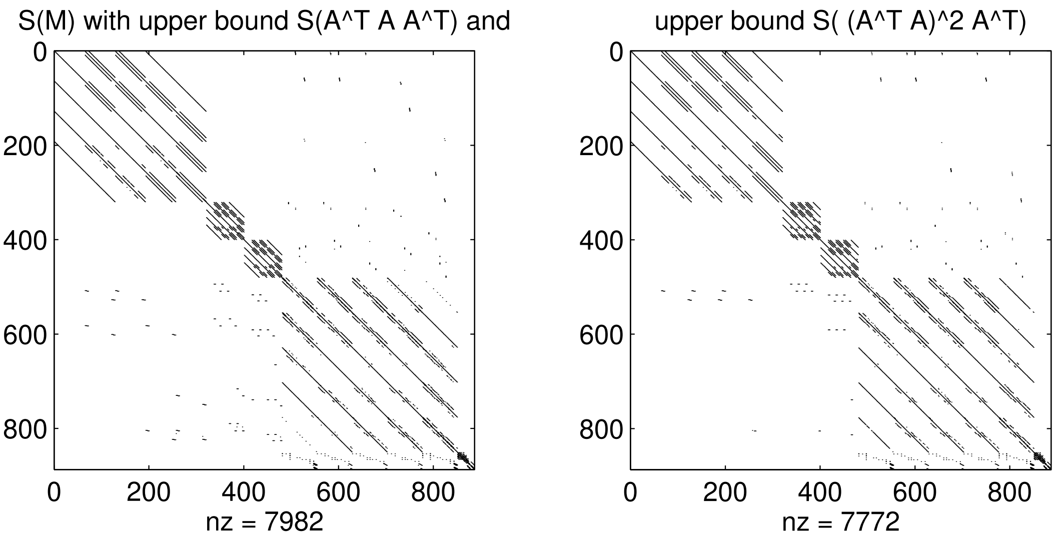

Figure 5.: $S(M)$ for SPAI with upper bounds $S(A^TAA^T)$ and $S((A^TA)^2A^T)$.

Note, that unrestricted SPAI and SPAI with upper bound $S((A^TA)^2A^T)$ lead to the same matrix M, which is very similar to the matrix M computed with upper bound $S(A^TAA^T)$. Furthermore we see that very soon the patterns $S((A^TA)^mA^T)$ are getting nearly dense. Hence it is not possible to choose e.g. $S(A^TAA^T)$ as an exact pattern for the sparse approximate inverse matrix M.

We also can compare the different approximations by considering the residual in the Frobenius norm. Here, the SPAI approxiamtions show a much better behaviour than the truncated inverse matrices.

M	$\|A * M - E\|_F$	nnz(M)
$\|A^{-1}\| > 0.002$	90.4	7485
$\|A^{-1}\| > 0.0015$	188.0	11996
$\|A^{-1}\| > 0.001$	130.1	19984
$\|A^{-1}\| > 0.0005$	37.6	46296
$\|A^{-1}\| > 0.0001$	11.1	165650
SPAI full pattern	7.70	7772
SPAI $S(A^T)$	13.3	5737
SPAI $S(A^TAA^T)$	8.14	7982
SPAI $S((A^TA)^2A^T)$	7.70	7772

Table 1: Residual for different approximations

Next we compute sparse approximate inverses M by the SPAI algorithm for different matrices and different upper bounds for the pattern of M. Then we use M as preconditioner for solving linear equations. As iterative method we chose BiCGSTAB with stopping criterion $\|r_k\|/\|r_0\| \leq 10^{-6}$. As right hand sides we considered $b = (1, ..., 1)^T$, $b = (1, 0, ..., 0, 2)^T$, and a random vector b. In the following tables $nnz(M)$ denotes the number of nonzero entries in the computed sparse approximate inverse, $nnz(S)$ the number of nonzero entries in the prescribed upper bound pattern S, $nnz(r_j > \epsilon)$ the number of columns that do not satisfy $\|AM_j - e_j\| \leq \epsilon$. Note that it is only meaningful to use patterns with $nnz(S) << n^2$. The test matrices are a band Toeplitz matrix or they are taken from the Harwell-Boeing Collection.

We see that - using the results of the previous sections - it is possible to define sparse patterns S that allow a good approximation on A^{-1}. In all examples $S((A^TA)^mA^T)$ and $S((E + |A| + |A^T|)^m)$ for some $m \leq 3$ give a satisfactory, easy to compute, and sparse approximate upper bound pattern. In the case of a triangular matrix A the pattern $(E + |A|)^5$ is slightly better than the equations based on both A and A^T.

pattern S	k	$nnz(M)$	$\|AM-E\|_F$	$nnz(r_j > \epsilon)$	$nnz(S)$
full pattern	–	3484	7.90	1	250000
$(E+\|A\|)^k$	4	2490	9.08	495	2490
	5	2985	8.41	0	2985
$(A^TA)^kA^T$	1	1993	9.99	498	1996
	2	2986	8.44	3	2991
	3	2989	8.44	2	3984
$(E+\|A\|+\|A^T\|)^k$	1	1497	11.16	498	1498
	2	2988	8.42	2	2991
$(E+\|A\|+\|A^T\|)^kA^T$	1	1993	9.99	498	1996
	2	2986	8.44	3	2991

Table 2: $A = tridiag(-1, 1, 0)$, $n = 500$, $nnz(A) = 999$, $\epsilon = 0.4$, $to = 15$

pattern S	k	$nnz(M)$	$\|AM-E\|_F$	$nnz(S)$	It1	It2	It3
full pattern	–	4139	8.78	784996	34	33	36
$(E+\|A\|+\|A^T\|)^k$	0	886	17.98	886	409	312	399
	1	4380	13.35	5970	139	118	132
	2	6667	11.52	20850	58	51	52
	3	4186	9.00	51456	37	31	36
	4	4131	8.78	101672	34	33	36
$((E+\Delta A)^k\Delta$	1	7288	14.44	30628	$*$	$*$	$*$
	2	6155	13.03	110280	$*$	$*$	$*$
without prec.	-	-	-	-	840	591	986

Table 3: ORRSIR2, $n = 886$, $nnz(A) = 5970$, $\epsilon = 0.4$, $to = 15$

pattern S	k	$nnz(M)$	$\|AM - E\|$	$\#(r_j > \epsilon)$	$nnz(S)$	It1	It2	It3
full pattern	–	7268	11.07	439	262144	477	261	444
$\|A\|^k$	6	6312	13.28	439	44028	$*$	$*$	$*$
$(E + \|A\|)^k$	2	2852	19.70	496	5902	$*$	$*$	$*$
	3	7518	11.59	483	12430	1002	1091	2142
	4	7309	11.06	441	22034	458	416	480
$(A^T A)^k A^T$	0	1970	14.54	497	1976	153	326	328
	1	7302	11.23	442	14118	591	325	539
	2	7300	11.05	439	48548	537	337	530
$(E + \|A\| + \|A^T\|)^k$	2	7174	12.71	456	13014	179	222	207
	3	7317	11.11	439	29158	385	405	586
$(E + \|A\| + \|A^T\|)^k A^T$	1	7170	12.77	458	10806	351	393	240
	2	7317	11.14	439	27458	364	582	344
$A^T(E + \|A\| + \|A^T\|)^k$	1	7162	12.79	458	10806	407	249	212
	2	7317	11.15	440	27458	262	435	352
$(E + \Delta A)^k \Delta$	1	7317	11.13	444	23896	566	562	419
	2	7300	11.05	442	68904	477	261	444

Table 4: $GRE512$, $n = 512$, $nnz(A) = 1976$, $\epsilon = 0.4$, $to = 15$

pattern S	k	$nnz(M)$	$\|AM - E\|_F$	$nnz(S)$	It1	It2	It3
full pattern	–	6417	6.09	250000	366	100	82
$\|A\|^k$	1	61497	9.37	1497	991	292	232
	2	2984	7.88	2990	727	189	152
	3	4463	6.97	4478	520	139	109
$(A^T A)^k A^T$	0	1497	12.11	1497	$*$	731	589
	1	4950	6.92	4975	631	148	121
	2	6417	6.09	7936	366	100	82
$(\|A\| + \|A^T\|)^k$	1	2489	8.21	2494	730	210	165
	2	4461	7.01	4480	551	139	114
	3	5443	6.32	6458	393	110	91
$(\|A\| + \|A^T\|)^k A^T$	1	3970	8.05	3984	781	195	155
	2	5441	6.68	5964	532	124	104
$A^T(\|A\| + \|A^T\|)^k$	1	3970	8.05	3984	781	195	155
	2	5441	6.68	5964	532	124	104
$((E + \Delta A)^k \Delta$	1	5443	6.32	5964	393	110	91

Table 5: $pentdiag(0, -1, 2, 0, -1)$, $n = 500$, $nnz(A) = 1497$, $\epsilon = 0.3$, $to = 15$

pattern S	k	Mflops	$nnz(M_{556})$	$\|AM_{556} - e_{556}\|$	$nnz(S_{556})$
full pattern	–	0.17	54	0.25	822
$(E + \|A\|)^k = \|A\|^k$	7	0.40	80	0.97	651
	8	0.17	54	0.25	766
$(A^T A)^k A^T$	2	0.28	66	0.28	366
	3	0.17	54	0.25	716
$(E + \|A\| + \|A^T\|)^k$	2	0.64	80	0.998	311
	3	0.18	54	0.25	760
$(E + \|A\| + \|A^T\|)^k A^T$	2	0.06	31	0.33	200
	3	0.18	54	0.25	684
$A^T(E + \|A\| + \|A^T\|)^k$	3	0.48	54	0.25	806
$(E + \Delta A)^k \Delta$	1	0.29	66	0.30	405
	2	0.18	54	0.25	816

Table 6: $BP200$, column 556, $n = 822$, $nnz(A(:, 556)) = 5$, $nnz(A) = 3803$, $\epsilon = 0.4$, $to = 80$

This last example shows that with an appropriate a-priori pattern sometimes the resulting sparse approximate column M_k can be computed faster and/or have less entries.

References

[1] O. Axelsson, *Iterative Solution Methods* (Cambridge University Press, 1994).

[2] S.T. Barnard and R.L. Clay, A portable implementation of the SPAI preconditioner in ISIS++, in *Proceedings of the Eight SIAM Conference on Parallel Processing for Scientific Computing,* M. Heath et al., eds (SIAM, Philadelphia, 1997).

[3] R. Barrett, M. Berry, T. Chan, J. Demmel, J. Donato, J Dongarra, V. Eijkhout, R. Pozo, C. Romine, H. van der Vorst, *Templates for the solution of linear systems: building blocks for iterative methods*, (SIAM, Philadelphia, 1994).

[4] M. Benzi and M. Tuma, Numerical Experiments with two Approximate Inverse Preconditioners.

[5] E. Chow and Y. Saad, Approximate Inverse Preconditioners for general sparse matrices, Research Report UMSI 94/101, University of Minnesota Supercomputing Institute, Minneapolis, Minnesota, 1994.

[6] J.D.F. Cosgrove, J.C. Diaz, and A. Griewank, Approximate inverse preconditioning for sparse linear systems, *Intl. J. Comp. Math.* **44** (1992) 91-110.

[7] J.R. Gilbert, Predicting structure in sparse matrix computations, *SIAM J. Matrix Anal. Appl.* **15**(1) (1994) 62-79.

[8] N.I.M. Gould and J.A. Scott, On approximate-inverse preconditioners, Technical Report RAL 95-026, Rutherford Appleton Laboratory, Chilton, England, 1995.

[9] M. Grote and T. Huckle, Parallel preconditioning with sparse approximate inverses, *SIAM J. Sci. Comput.* **18** (3) (1997) 838–853.

[10] T. Huckle, Efficient Computation of Sparse Approximate Inverses, TUM Technical Report TUM-19608 SFB 342/04/96 A, submitted to J. Numer. Linear Alg.

[11] T. Huckle, Sparse Approximate Inverses for Preconditioning of Linear Equations, Conferentie van Numeriek Wiskundigen, Woudschoten, Zeist, The Netherlands (1996).

[12] I.E. Kaporin, New Convergence Results and Preconditioning Strategies for the Conjugate Gradient Method, *Num. Lin. Alg.* **1** (2) (1994) 179–210.

SFB 342: Methoden und Werkzeuge für die Nutzung paralleler Rechnerarchitekturen

bisher erschienen :

Reihe A

Liste aller erschienenen Berichte von 1990-1994
auf besondere Anforderung

342/01/95 A Hans-Joachim Bungartz: Higher Order Finite Elements on Sparse Grids
342/02/95 A Tao Zhang, Seonglim Kang, Lester R. Lipsky: The Performance of Parallel Computers: Order Statistics and Amdahl's Law
342/03/95 A Lester R. Lipsky, Appie van de Liefvoort: Transformation of the Kronecker Product of Identical Servers to a Reduced Product Space
342/04/95 A Pierre Fiorini, Lester R. Lipsky, Wen-Jung Hsin, Appie van de Liefvoort: Auto-Correlation of Lag-k For Customers Departing From Semi-Markov Processes
342/05/95 A Sascha Hilgenfeldt, Robert Balder, Christoph Zenger: Sparse Grids: Applications to Multi-dimensional Schrödinger Problems
342/06/95 A Maximilian Fuchs: Formal Design of a Model-N Counter
342/07/95 A Hans-Joachim Bungartz, Stefan Schulte: Coupled Problems in Microsystem Technology
342/08/95 A Alexander Pfaffinger: Parallel Communication on Workstation Networks with Complex Topologies
342/09/95 A Ketil Stølen: Assumption/Commitment Rules for Data-flow Networks - with an Emphasis on Completeness
342/10/95 A Ketil Stølen, Max Fuchs: A Formal Method for Hardware/Software Co-Design
342/11/95 A Thomas Schnekenburger: The ALDY Load Distribution System
342/12/95 A Javier Esparza, Stefan Römer, Walter Vogler: An Improvement of McMillan's Unfolding Algorithm
342/13/95 A Stephan Melzer, Javier Esparza: Checking System Properties via Integer Programming
342/14/95 A Radu Grosu, Ketil Stølen: A Denotational Model for Mobile Point-to-Point Dataflow Networks
342/15/95 A Andrei Kovalyov, Javier Esparza: A Polynomial Algorithm to Compute the Concurrency Relation of Free-Choice Signal Transition Graphs
342/16/95 A Bernhard Schätz, Katharina Spies: Formale Syntax zur logischen Kernsprache der Focus-Entwicklungsmethodik
342/17/95 A Georg Stellner: Using CoCheck on a Network of Workstations
342/18/95 A Arndt Bode, Thomas Ludwig, Vaidy Sunderam, Roland Wismüller: Workshop on PVM, MPI, Tools and Applications
342/19/95 A Thomas Schnekenburger: Integration of Load Distribution into ParMod-C

Reihe A

342/20/95 A Ketil Stølen: Refinement Principles Supporting the Transition from Asynchronous to Synchronous Communication

342/21/95 A Andreas Listl, Giannis Bozas: Performance Gains Using Subpages for Cache Coherency Control

342/22/95 A Volker Heun, Ernst W. Mayr: Embedding Graphs with Bounded Treewidth into Optimal Hypercubes

342/23/95 A Petr Jančar, Javier Esparza: Deciding Finiteness of Petri Nets up to Bisimulation

342/24/95 A M. Jung, U. Rüde: Implicit Extrapolation Methods for Variable Coefficient Problems

342/01/96 A Michael Griebel, Tilman Neunhoeffer, Hans Regler: Algebraic Multigrid Methods for the Solution of the Navier-Stokes Equations in Complicated Geometries

342/02/96 A Thomas Grauschopf, Michael Griebel, Hans Regler: Additive Multilevel-Preconditioners based on Bilinear Interpolation, Matrix Dependent Geometric Coarsening and Algebraic-Multigrid Coarsening for Second Order Elliptic PDEs

342/03/96 A Volker Heun, Ernst W. Mayr: Optimal Dynamic Edge-Disjoint Embeddings of Complete Binary Trees into Hypercubes

342/04/96 A Thomas Huckle: Efficient Computation of Sparse Approximate Inverses

342/05/96 A Thomas Ludwig, Roland Wismüller, Vaidy Sunderam, Arndt Bode: OMIS — On-line Monitoring Interface Specification

342/06/96 A Ekkart Kindler: A Compositional Partial Order Semantics for Petri Net Components

342/07/96 A Richard Mayr: Some Results on Basic Parallel Processes

342/08/96 A Ralph Radermacher, Frank Weimer: INSEL Syntax-Bericht

342/09/96 A P.P. Spies, C. Eckert, M. Lange, D. Marek, R. Radermacher, F. Weimer, H.-M. Windisch: Sprachkonzepte zur Konstruktion verteilter Systeme

342/10/96 A Stefan Lamberts, Thomas Ludwig, Christian Röder, Arndt Bode: PFSLib – A File System for Parallel Programming Environments

342/11/96 A Manfred Broy, Gheorghe Ştefănescu: The Algebra of Stream Processing Functions

342/12/96 A Javier Esparza: Reachability in Live and Safe Free-Choice Petri Nets is NP-complete

342/13/96 A Radu Grosu, Ketil Stølen: A Denotational Model for Mobile Many-to-Many Data-flow Networks

342/14/96 A Giannis Bozas, Michael Jaedicke, Andreas Listl, Bernhard Mitschang, Angelika Reiser, Stephan Zimmermann: On Transforming a Sequential SQL-DBMS into a Parallel One: First Results and Experiences of the MIDAS Project

342/15/96 A Richard Mayr: A Tableau System for Model Checking Petri Nets with a Fragment of the Linear Time μ -Calculus

Reihe A

342/16/96 A	Ursula Hinkel, Katharina Spies: Anleitung zur Spezifikation von mobilen, dynamischen Focus-Netzen
342/17/96 A	Richard Mayr: Model Checking PA-Processes
342/18/96 A	Michaela Huhn, Peter Niebert, Frank Wallner: Put your Model Checker on Diet: Verification on Local States
342/01/97 A	Tobias Müller, Stefan Lamberts, Ursula Maier, Georg Stellner: Evaluierung der Leistungsf”ahigkeit eines ATM-Netzes mit parallelen Programmierbibliotheken
342/02/97 A	Hans-Joachim Bungartz and Thomas Dornseifer: Sparse Grids: Recent Developments for Elliptic Partial Differential Equations
342/03/97 A	Bernhard Mitschang: Technologie f”ur Parallele Datenbanken - Bericht zum Workshop
342/04/97 A	nicht erschienen
342/05/97 A	Hans-Joachim Bungartz, Ralf Ebner, Stefan Schulte: Hierarchische Basen zur effizienten Kopplung substrukturierter Probleme der Strukturmechanik
342/06/97 A	Hans-Joachim Bungartz, Anton Frank, Florian Meier, Tilman Neunhoeffer, Stefan Schulte: Fluid Structure Interaction: 3D Numerical Simulation and Visualization of a Micropump
342/07/97 A	Javier Esparza, Stephan Melzer: Model Checking LTL using Constraint Programming
342/08/97 A	Niels Reimer: Untersuchung von Strategien für verteiltes Last- und Ressourcenmanagement
342/09/97 A	Markus Pizka: Design and Implementation of the GNU INSEL-Compiler gic
342/10/97 A	Manfred Broy, Franz Regensburger, Bernhard Schätz, Katharina Spies: The Steamboiler Specification - A Case Study in Focus
342/11/97 A	Christine Röckl: How to Make Substitution Preserve Strong Bisimilarity
342/12/97 A	Christian B. Czech: Architektur und Konzept des Dycos-Kerns
342/13/97 A	Jan Philipps, Alexander Schmidt: Traffic Flow by Data Flow
342/14/97 A	Norbert Fröhlich, Rolf Schlagenhaft, Josef Fleischmann: Partitioning VLSI-Circuits for Parallel Simulation on Transistor Level
342/15/97 A	Frank Weimer: DaViT: Ein System zur interaktiven Ausführung und zur Visualisierung von INSEL-Programmen
342/16/97 A	Niels Reimer, Jürgen Rudolph, Katharina Spies: Von FOCUS nach INSEL - Eine Aufzugssteuerung
342/17/97 A	Radu Grosu, Ketil Stølen, Manfred Broy: A Denotational Model for Mobile Point-to-Point Data-flow Networks with Channel Sharing
342/18/97 A	Christian Röder, Georg Stellner: Design of Load Management for Parallel Applications in Networks of Heterogenous Workstations
342/19/97 A	Frank Wallner: Model Checking LTL Using Net Unfoldings

Reihe A

342/20/97 A	Andreas Wolf, Andreas Kmoch: Einsatz eines automatischen Theorembeweisers in einer taktikgesteuerten Beweisumgebung zur Lösung eines Beispiels aus der Hardware-Verifikation – Fallstudie –
342/21/97 A	Andreas Wolf, Marc Fuchs: Cooperative Parallel Automated Theorem Proving
342/22/97 A	T. Ludwig, R. Wismüller, V. Sunderam, A. Bode: OMIS - On-line Monitoring Interface Specification (Version 2.0)
342/23/97 A	Stephan Merkel: Verification of Fault Tolerant Algorithms Using PEP
342/24/97 A	Manfred Broy, Max Breitling, Bernhard Schätz, Katharina Spies: Summary of Case Studies in Focus - Part II
342/25/97 A	Michael Jaedicke, Bernhard Mitschang: A Framework for Parallel Processing of Aggregat and Scalar Functions in Object-Relational DBMS
342/26/97 A	Marc Fuchs: Similarity-Based Lemma Generation with Lemma-Delaying Tableau Enumeration
342/27/97 A	Max Breitling: Formalizing and Verifying TimeWarp with FOCUS
342/28/97 A	Peter Jakobi, Andreas Wolf: DBFW: A Simple DataBase FrameWork for the Evaluation and Maintenance of Automated Theorem Prover Data (incl. Documentation)
342/29/97 A	Radu Grosu, Ketil Stølen: Compositional Specification of Mobile Systems
342/01/98 A	A. Bode, A. Ganz, C. Gold, S. Petri, N. Reimer, B. Schiemann, T. Schnekenburger (Herausgeber): "'Anwendungsbezogene Lastverteilung'", ALV'98
342/02/98 A	Ursula Hinkel: Home Shopping - Die Spezifikation einer Kommunikationsanwendung in FOCUS
342/03/98 A	Katharina Spies: Eine Methode zur formalen Modellierung von Betriebssystemkonzepten
342/04/98 A	Stefan Bischof, Ernst-W. Mayr: On-Line Scheduling of Parallel Jobs with Runtime Restrictions
342/05/98 A	St. Bischof, R. Ebner, Th. Erlebach: Load Balancing for Problems with Good Bisectors and Applications in Finite Element Simulations: Worst-case Analysis and Practical Results
342/06/98 A	Giannis Bozas, Susanne Kober: Logging and Crash Recovery in Shared-Disk Database Systems
342/07/98 A	Markus Pizka: Distributed Virtual Address Space Management in the MoDiS-OS
342/08/98 A	Niels Reimer: Strategien für ein verteiltes Last- und Ressourcenmanagement
342/09/98 A	Javier Esparza, Editor: Proceedings of INFINITY'98
342/10/98 A	Richard Mayr: Lossy Counter Machines

SFB 342 : Methoden und Werkzeuge für die Nutzung paralleler Rechnerarchitekturen

Reihe B

342/1/90 B	Wolfgang Reisig: Petri Nets and Algebraic Specifications
342/2/90 B	Jörg Desel: On Abstraction of Nets
342/3/90 B	Jörg Desel: Reduction and Design of Well-behaved Free-choice Systems
342/4/90 B	Franz Abstreiter, Michael Friedrich, Hans-Jürgen Plewan: Das Werkzeug runtime zur Beobachtung verteilter und paralleler Programme
342/1/91 B	Barbara Paech1: Concurrency as a Modality
342/2/91 B	Birgit Kandler, Markus Pawlowski: SAM: Eine Sortier- Toolbox - Anwenderbeschreibung
342/3/91 B	Erwin Loibl, Hans Obermaier, Markus Pawlowski: 2. Workshop über Parallelisierung von Datenbanksystemen
342/4/91 B	Werner Pohlmann: A Limitation of Distributed Simulation Methods
342/5/91 B	Dominik Gomm, Ekkart Kindler: A Weakly Coherent Virtually Shared Memory Scheme: Formal Specification and Analysis
342/6/91 B	Dominik Gomm, Ekkart Kindler: Causality Based Specification and Correctness Proof of a Virtually Shared Memory Scheme
342/7/91 B	W. Reisig: Concurrent Temporal Logic
342/1/92 B	Malte Grosse, Christian B. Suttner: A Parallel Algorithm for Set-of-Support Christian B. Suttner: Parallel Computation of Multiple Sets-of-Support
342/2/92 B	Arndt Bode, Hartmut Wedekind: Parallelrechner: Theorie, Hardware, Software, Anwendungen
342/1/93 B	Max Fuchs: Funktionale Spezifikation einer Geschwindigkeitsregelung
342/2/93 B	Ekkart Kindler: Sicherheits- und Lebendigkeitseigenschaften: Ein Literaturüberblick
342/1/94 B	Andreas Listl; Thomas Schnekenburger; Michael Friedrich: Zum Entwurf eines Prototypen für MIDAS