RWTH Aachen

Department of Computer Science
Technical Report

Toward Low Static Memory Jacobian Accumulation

Ebadollah Varnik, Uwe Naumann and Andrew Lyons

ISSN 0935–3232 · Aachener Informatik Berichte · AIB-2006-04

RWTH Aachen · Department of Computer Science · March 2006

Toward Low Static Memory Jacobian Accumulation

Ebadollah Varnik, Uwe Naumann[1] and and Andrew Lyons[2]

[1] LuFG Informatik XII, Software and Tools for Computational Engineering
Department of Computer Science RWTH Aachen University, 52056 Aachen, Germany
Email:{`varnik|naumann`}`@stce.rwth-aachen.de`
[2] Department of Mathematics, Vanderbilt University
VU Station B Box 2895, Nashville TN 37235, USA
Email:`lyonsam@gmail.com`

Abstract. Derivatives are essential ingredients of a wide range of numerical algorithms. We focus on the accumulation of Jacobian matrices by Gaussian elimination on a sparse implementation of the extended Jacobian. A symbolic algorithm is proposed to determine the fill-in. The first version of the new algorithm results in a speedup of five compared to the elimination algorithm that does not exploit sparsity. On the given computer architecture we are able to handle problems with roughly four times the original size.

1 Introduction

The context of this paper is *automatic differentiation* [1, 3, 2] of numerical programs. We consider vector functions

$$F : I\!R^n \supseteq D \rightarrow I\!R^m, \quad \mathbf{y} = F(\mathbf{x}) \quad , \tag{1}$$

that map a vector $\mathbf{x} \equiv (x_i)_{i=1,\ldots,n}$ of *independent* variables onto a vector $\mathbf{y} \equiv (y_j)_{j=1,\ldots,m}$ of *dependent* variables. We assume that F has been implemented as a computer program. Hence, it can be decomposed into a sequence of p single assignments of the value of scalar *elemental* functions φ_i to unique *intermediate* variables v_j. This *code list* of F is given as

$$(I\!R \ni)\ v_j = \varphi_j(v_i)_{i \prec j} \quad , \tag{2}$$

where $j = n+1, \ldots, q$ and $q = n + p + m$. The binary relation $i \prec j$ denotes a direct dependence of v_j on v_i. So, $P_j = \{i : i \prec j\}$ is the index set of the arguments of φ_j. Similarly, $S_j = \{i : j \prec i\}$ is the index set of the elemental functions that have v_j as an argument. The variables $\mathbf{v} = (v_i)_{i=1,\ldots,q}$ are partitioned into the sets X containing the *independent* variables $(\mathbf{v}_i)_{i=1,\ldots,n}$, Y containing the *dependent* variables $(v_i)_{i=n+p+1,\ldots,q}$, and Z containing the intermediate variables $(v_i)_{i=n+1,\ldots,n+p}$. The code list of F can be represented as a directed acyclic *computational graph* $G = G(F) = (V, E)$ with integer vertices $V = \{i : i \in \{1, \ldots, q\}\}$ and edges $(i, j) \in E$ if and only if $i \prec j$. Moreover, $V = X \cup Z \cup Y$, where $X = \{1, \ldots, n\}$, $Z = \{n+1, \ldots, n+p\}$, and $Y = \{n+p+1, \ldots, q\}$. Hence, X, Y, and Z are mutually disjoint. We distinguish between *independent* ($i \in X$), *intermediate* ($i \in Z$), and *dependent* ($i \in Y$) vertices. Under the assumption that all elemental functions are continuously differentiable in some neighborhood of their arguments all edges (i, j) can be labeled with the partial derivatives $c_{j,i} \equiv \frac{\partial v_j}{\partial v_i}$ of v_j w.r.t. v_i. This labeling yields the *linearized* computational graph G of F. From now on we use the notation G to refer to the linearized computational graph. Equation (2) can be written as a system of nonlinear equation $C(\mathbf{v})$ [5] as follows.

$$\varphi_j(v_i)_{i \prec j} - v_j = 0 \quad \text{for } j = n+1, \ldots, q \quad . \tag{3}$$

Differentiation with respect to $\mathbf{v}$ leads to

$$C' = C'(\mathbf{v}) \equiv (c'_{j,i})_{i,j=1,\ldots,q} = \begin{cases} c_{j,i} & \text{if } i \prec j \\ -1 & \text{if } i = j \\ 0 & \text{otherwise} \quad . \end{cases} \tag{4}$$

The *extended Jacobian* C' is lower triangular. Its rows and columns are enumerated as $j, i = 1, \ldots, q$. Row j of C' corresponds to vertex j of G and contains the partial derivatives $c_{j,k}$ of vertex j w.r.t. all of its predecessors $k \in P_j$. In the following we refer to a row i as *independent* for $i \in \{1, \ldots, n\}$, as *intermediate* for $i \in \{n+1, \ldots, n+p\}$, and as *dependent* if $i \in \{n+p+1, \ldots, q\}$.

The focus of this paper is on reducing the static memory requirement of Jacobian accumulation by Gaussian elimination on C'. We aim to reuse memory freed due to so called *fill-out* to store *fill-in* generated during the elimination process. The structure of the paper is as follows: In Section 2 we review *compressed row storage* (CRS) for storing C'. In Section 3 we introduce a *symbolic* algorithm that uses a sparse bit pattern to detect fill-in and fill-out. Furthermore, we present first ideas on how to make use of fill-out to further reduce memory requirement. Section 5 shows some runtime and memory analysis.

1.1 Elimination Techniques

The *Jacobian matrix* (or simply *Jacobian*) of F as defined in Equation (1) at point $\mathbf{x}_0$ is defined as follows:

$$(I\!R^{m \times n} \ni) \; F' = F'(\mathbf{x}_0) \equiv \left(\frac{\partial y_i}{\partial x_j}(\mathbf{x}_0) \right)_{j=1,\ldots,n}^{i=1,\ldots,m} \quad .$$

F' can be obtained by eliminating all intermediate vertices $j \in Z$ from G as introduced in [6]. Each predecessor $i \in P_j$ of j is connected with all successors $k \in S_j$. If $(i,k) \notin E$, then it has to be generated and labeled with $c_{k,i} := c_{k,j} \cdot c_{j,i}$. Otherwise the value of $c_{k,i}$ is updated as $c_{k,i} := c_{k,i} + c_{k,j} \cdot c_{j,i}$. In the former case we say that fill-in is generated whereas *absorption* takes place in the latter. The elimination of vertex j can be understood as some sort of *Gaussian* elimination of all non-zero entries in row/column j of C'. Therefore one has to find all those rows k with $j \prec k$. In order to eliminate row/column j we perform the following transformation on C'.

Definition 1 (Row/Column Elimination in C')

$$c_{k,i} := c_{k,i} + c_{k,j} \cdot c_{j,i} \quad \forall i \prec j \wedge \forall k : j \prec k \tag{5}$$

$$c_{j,i} := 0 \quad \forall i \prec j \tag{6}$$

$$c_{k,j} := 0 \quad \forall k : j \prec k \tag{7}$$

$$c_{j,j} := 0 \quad . \tag{8}$$

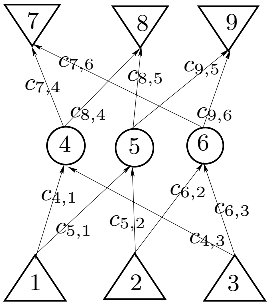
Note that $c_{k,i} = 0$ if $i \not\prec k$. The new partial derivatives of v_k, $j \prec k$, with respect to v_i, $i \prec j$, are computed by applying the chain rule in Equation (5). Hence, any sensitivities of the v_k on v_j as well as of v_j on any of the v_i are removed in Equation (6) and Equation (7), respectively. Fill-out is generated. Setting the diagonal entry $c_{j,j}$ to zero in Equation (8) leads to the removal of the j-th row and column in C'. If $c_{k,i} = 0$ then Equation (5) leads to fill-in otherwise it yields absorption.

1.2 Example

Consider the vector function $F : R^3 \rightarrow R^3$ whose code list is given in Figure 1(a). The corresponding G and C' are shown in Figure 1 (b) and (c), respectively. The symbols $\triangle$ represent independent, $\bigtriangledown$ dependent, and $\bigcirc$ intermediate vertices. Consider row 5 in Figure 1 (c) containing $c_{5,1}$ and $c_{5,2}$. These are labels of incoming edges $(1,5)$ and $(2,5)$ of vertex 5 in Figure 1 (b). Column 5 contains the partial derivatives $c_{8,5}$ and $c_{9,5}$ that are the labels of outgoing edges $(5,8)$ and $(5,9)$ of vertex 5. In the context of symbolic elimination we are merely interested in the sparsity structure of C'. Hence, $\times$ represents fill-in, $\bigcirc$ represents fill-out, and blanks represent zeros in C'. Eliminating $c_{5,1}$ in C' is equivalent to *front-elimination* [7] of $(1,5)$ as shown in Figure 2 (a). Fill-in is generated as $c_{8,1}$ $[(1,8)]$ and $c_{9,1}$ $[(1,9)]$ since rows [vertices] 8 and 9 have non-zeros [incoming edges] in [from] column [vertex] 5. The elimination of the row/column [vertex] 5 in C' $[G]$ can be done by elimination [front-elimination] of all non-zeros [incoming edges] in [to] row/column [vertex] 5. The resulting fill-in, namely $c_{8,1}$, $c_{8,2}$, $c_{9,1}$, and $c_{9,2}$ $[(1,8)$, $(2,8)$, $(1,9)$, and $(2,9)]$ in C' $[G]$ is shown in Figure 3 (b) [(a)]. A total of $p!$ different row [vertex] elimination orderings in C' $[G']$ are possible. In this paper we focus on *reverse elimination* $(n+p, \cdots, n+1)$. Hence, the Jacobian F' [the bipartite graph G'] is derived from C' $[G]$ by elimination of all intermediate rows [vertices] in order $(6,5,4)$. The result is shown in Figure 4 (b) [(a)].

$v_1 = x_1$
$v_2 = x_2$
$v_3 = x_3$
$v_4 = v_1 * v_3$
$v_5 = v_1 * v_2$
$v_6 = v_2 * v_3$
$v_7 = v_4 * v_6$
$v_8 = v_4 * v_5$
$v_9 = v_5 * v_6$

(a)



(b)

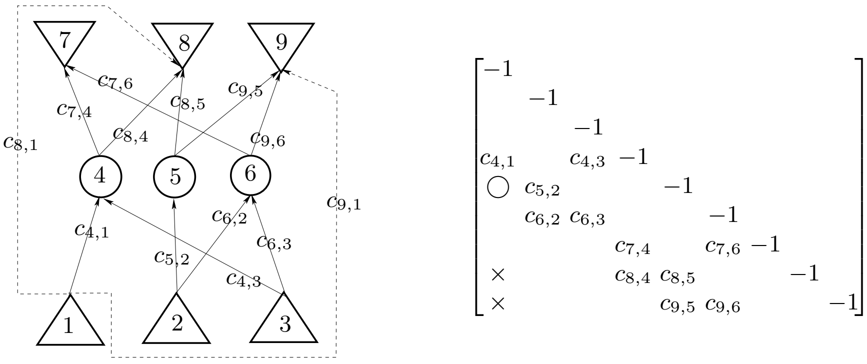
$$\begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ c_{4,1} & 0 & c_{4,3} & -1 & 0 & 0 & 0 & 0 & 0 \\ c_{5,1} & c_{5,2} & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & c_{6,2} & c_{6,3} & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & c_{7,4} & 0 & c_{7,6} & -1 & 0 & 0 \\ 0 & 0 & 0 & c_{8,4} & c_{8,5} & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & c_{9,5} & c_{9,6} & 0 & 0 & -1 \end{bmatrix}$$

(c)

Fig. 1. Code list (a); linearized computational graph G (b); C' (c) of F.

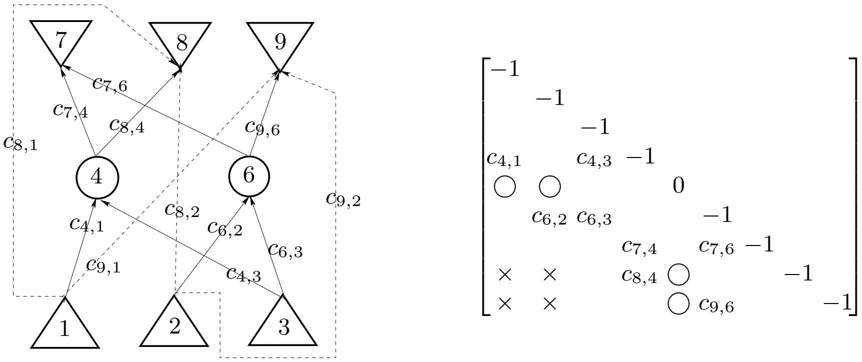
2 The Problem

In order to exploit the sparsity of C' we use compressed row storage [4]. Fill-in and fill-out may result in additional memory allocation and lost memory in CRS,



(a) (b)

Fig. 2. G $[C']$ after front-elimination [elimination] of $(1,5)$ $[c_{5,1}]$ (a) [(b)].



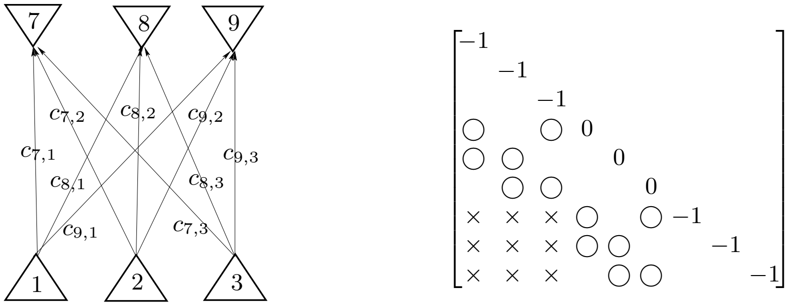
(a) (b)

Fig. 3. G $[C']$ after elimination of vertex [row/column] 5 (a) [(b)].

respectively. As mentioned above F' can be computed from C' by reverse elimination. Different elimination orderings may yield different amounts of fill-in and fill-out. Finding an optimal (or near optimal) elimination ordering is the subject of ongoing work. In this paper we are merely interested in the computation of F' while exploiting the sparsity of C' by using CRS. Our objective is to allocate (once) static memory for accumulation of F' from a given C' in CRS format by reverse elimination. The extended Jacobian C' of F as in Figure 1 (a) has the following CRS representation:

$$\begin{aligned}
\alpha =&[c_{4,1}, c_{4,3}, c_{5,1}, c_{5,2}, c_{6,2}, c_{6,3}, c_{7,4}, c_{7,6}, c_{8,4}, c_{8,5}, c_{9,5}, c_{9,6}] \\
\kappa =&[1, 3, 1, 2, 2, 3, 4, 6, 4, 5, 5, 6] \\
\rho =&[1, 1, 1, 1, 3, 5, 7, 9, 11, 13] \quad .
\end{aligned}$$

The CRS currently does not contain memory for potential fill-in. Therefore we need a fill-in detection procedure to allocate enough memory in our CRS structure. The corresponding symbolic elimination algorithm (see Section 3) yields the following CRS structure:



(a) (b)

Fig. 4. Bipartite graph G' (a) and the corresponding structure of C' (b) after reverse elimination; The Jacobian is the 3×3 matrix in the lower left corner of C' after the elimination procedure.

$$
\begin{aligned}
\alpha =&[c_{4,1}, c_{4,3}, c_{5,1}, c_{5,2}, c_{6,2}, c_{6,3}, 0, 0, 0, c_{7,4}, c_{7,6}, 0, 0, 0, c_{8,4}, c_{8,5}, 0, 0, 0, c_{9,5}, c_{9,6}]\\
\kappa =&[1, 3, 1, 2, 2, 3, 1, 2, 3, 4, 6, 1, 2, 3, 4, 5, 1, 2, 3, 5, 6]\\
\rho =&[1, 1, 1, 1, 3, 5, 7, 12, 17, 22] \quad .
\end{aligned}
$$

Elimination of 6 yields :

$$
\begin{aligned}
\alpha =&[c_{4,1}, c_{4,3}, c_{5,1}, c_{5,2}, \mathbf{0}, \mathbf{0}, 0, c_{7,2}, c_{7,3}, c_{7,4}, \mathbf{0}, 0, 0, 0, c_{8,4}, c_{8,5}, 0, c_{9,2}, c_{9,3}, c_{9,5}, \mathbf{0}]\\
\kappa =&[1, 3, 1, 2, -\mathbf{1}, -\mathbf{1}, 1, 2, 3, 4, -\mathbf{1}, 1, 2, 3, 4, 5, 1, 2, 3, 5, -\mathbf{1}]\\
\rho =&[1, 1, 1, 1, 3, -\mathbf{1}, 12, 17, 22]
\end{aligned}
$$

where

$$
\begin{aligned}
&c_{7,2} := c_{6,2} \cdot c_{7,6}; \quad c_{7,3} := c_{6,3} \cdot c_{7,6};\\
&c_{9,2} := c_{6,2} \cdot c_{9,6}; \quad c_{9,3} := c_{6,3} \cdot c_{9,6} \quad .
\end{aligned}
$$

Elimination of 5 yields :

$$
\begin{aligned}
\alpha =&[c_{4,1}, c_{4,3}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}, 0, c_{7,2}, c_{7,3}, c_{7,4}, \mathbf{0}, c_{8,1}, c_{8,2}, 0, c_{8,4}, \mathbf{0}, c_{9,1}, c_{9,2}, c_{9,3}, \mathbf{0}, \mathbf{0}]\\
\kappa =&[1, 3, -\mathbf{1}, -\mathbf{1}, -\mathbf{1}, -\mathbf{1}, 1, 2, 3, 4, -\mathbf{1}, 1, 2, 3, 4, -\mathbf{1}, 1, 2, 3, -\mathbf{1}, -\mathbf{1}]\\
\rho =&[1, 1, 1, 1, -\mathbf{1}, -\mathbf{1}, 12, 17, 22]
\end{aligned}
$$

where

$$
\begin{aligned}
&c_{8,1} := c_{5,1} \cdot c_{8,5}; \quad c_{8,2} := c_{5,2} \cdot c_{8,5};\\
&c_{9,1} := c_{5,1} \cdot c_{9,5}; \quad c_{9,2} := c_{9,2} + c_{5,2} \cdot c_{9,5} \quad .
\end{aligned}
$$

Elimination of 4 yields :

$$
\begin{aligned}
\alpha =&[\mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}, c_{7,1}, c_{7,2}, c_{7,3}, \mathbf{0}, \mathbf{0}, c_{8,1}, c_{8,2}, c_{8,3}, \mathbf{0}, \mathbf{0}, c_{9,1}, c_{9,2}, c_{9,3}, \mathbf{0}, \mathbf{0}]\\
\kappa =&[-\mathbf{1}, -\mathbf{1}, -\mathbf{1}, -\mathbf{1}, -\mathbf{1}, -\mathbf{1}, 1, 2, 3, -\mathbf{1}, -\mathbf{1}, 1, 2, 3, -\mathbf{1}, -\mathbf{1}, 1, 2, 3, -\mathbf{1}, -\mathbf{1}]\\
\rho =&[1, 1, 1, -\mathbf{1}, -\mathbf{1}, -\mathbf{1}, 12, 17, 22]
\end{aligned}
$$

where

$$c_{7,1} := c_{4,1} \cdot c_{7,4}; \quad c_{7,3} := c_{7,3} + c_{4,3} \cdot c_{7,4};$$
$$c_{8,1} := c_{8,1} + c_{4,1} \cdot c_{8,4}; \quad c_{8,3} := c_{4,3} \cdot c_{8,4} \quad .$$

Bold **0**'s in α represent fill-out. Bold **-1**'s in κ and ρ mark eliminated entries and rows, respectively. For our example this approach leads to 9 additional memory allocations for fill-in and 12 fill-out's. Compared to the memory needed for storing the whole extended Jacobian we observe a decrease in memory requirement by less than a factor of 2. The savings are more substantial for larger problems as illustrated in Section 5. Moreover, we expect to decrease the memory further by reusing fill-out.

3 Symbolic Elimination Algorithm

Our symbolic fill-in detection algorithm uses a bit pattern $B = B(F)$ to hold the sparsity structure of C'. Figure 5 (a) shows the corresponding integer matrix for the extended Jacobian C' in Figure 1 (c). The binary representation is shown in Figure 5 (b). The symbolic algorithm is implemented in C++. Therefore we start counting with zero. Whenever we refer to the j-th row in B we mean the row with index $j - 1$.

$$\underset{\text{(a)}}{\begin{pmatrix} 0 \\ 0 \\ 0 \\ 40960 \\ \mathbf{49152} \\ 24576 \\ 5120 \\ 6144 \\ 3072 \end{pmatrix}} \qquad \underset{\text{(b)}}{\begin{pmatrix} 0\,0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0\,0\,0\,0\,0 \\ 1\,0\,1\,0\,0\,0\,0\,0\,0 \\ \mathbf{1\,1\,0\,0\,0\,0\,0\,0\,0} \\ 0\,1\,1\,0\,0\,0\,0\,0\,0 \\ 0\,0\,0\,1\,0\,1\,0\,0\,0 \\ 0\,0\,0\,1\,1\,0\,0\,0\,0 \\ 0\,0\,0\,0\,1\,1\,0\,0\,0 \end{pmatrix}}$$

Fig. 5. Bit pattern B as an integer matrix (a) and binary representation of C' (b).

Algorithm 1 (Symbolic Algorithm)
IN : B — bit pattern of C′
OUT: B — filled bit pattern after reverse elimination

```
[1] FOR $i = n + p - 1, \ldots, n$
[2]     FOR $j = q - 1, \ldots, i$
[3]       $k := i \gg 4;$
[4]       IF ( $B[j][k] \;\wedge\; 1 \ll (15 - i\%16)$ )
[5]         FOR $m = 0, \ldots, k$
[6]           $B[j][m] := B[j][m] \vee B[i][m];$
```

Consider the symbolic elimination of row 6 in Figure 5 (a) using Algorithm 1 with $i = 5$ and $j = 8$ in line *[1]* and *[2]*, respectively. The integer values corresponding to rows 6 and 9 are stored in column $k = 0$ (line *[3]*) with $B[5][0] = 24576$ and

$B[8][0] = 3072$. $6 \prec 9$ as in line *[4]* $24576 \wedge 2^{15-5} = \mathit{true}$. Hence, $B[8][0] = 27648 = 24576 \vee 3072$. Line *[5]* in Algorithm 1 performs the bitwise OR for all affected columns of B.

In the following we apply Algorithm 1 to the bit pattern of F shown in Figure 5 (a). The result is shown in Figure 6 (b). Symbolic elimination proceeds as follows:

$$\begin{pmatrix} 0 \\ 0 \\ 0 \\ 40960 \\ 49152 \\ 24576 \\ 5120 \\ 6144 \\ 3072 \end{pmatrix} \overset{elim(6)}{\longrightarrow} \begin{pmatrix} 0 \\ 0 \\ 0 \\ 40960 \\ 49152 \\ 24576 \\ \mathbf{29696} \\ 6144 \\ \mathbf{27648} \end{pmatrix} \overset{elim(5)}{\longrightarrow} \begin{pmatrix} 0 \\ 0 \\ 0 \\ 40960 \\ 49152 \\ 24576 \\ 29696 \\ \mathbf{55296} \\ \mathbf{60416} \end{pmatrix} \overset{elim(4)}{\longrightarrow} \begin{pmatrix} 0 \\ 0 \\ 0 \\ 40960 \\ 49152 \\ 24576 \\ \mathbf{62464} \\ \mathbf{63488} \\ 60416 \end{pmatrix}$$

where

$$\begin{aligned} &29696 = 2^{14} + 2^{13} + 5120; \quad 27648 = 2^{14} + 2^{13} + 3072; \\ &55296 = 2^{15} + 2^{14} + 6144; \quad 60416 = 2^{15} + 27648; \\ &62464 = 2^{15} + 29696; \quad 63488 = 2^{13} + 55296 \quad . \end{aligned}$$

$$\begin{pmatrix} 0 \\ 0 \\ 0 \\ 40960 \\ 49152 \\ 24576 \\ \mathbf{62464} \\ 63488 \\ 60416 \end{pmatrix} \qquad \begin{pmatrix} 0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0 \\ 1&0&1&0&0&0&0&0&0 \\ 1&1&0&0&0&0&0&0&0 \\ 0&1&1&0&0&0&0&0&0 \\ \mathbf{1}&\mathbf{1}&\mathbf{1}&\mathbf{1}&0&\mathbf{1}&0&0&0 \\ 1&1&1&1&1&0&0&0&0 \\ 1&1&1&0&1&1&0&0&0 \end{pmatrix}$$

(a) (b)

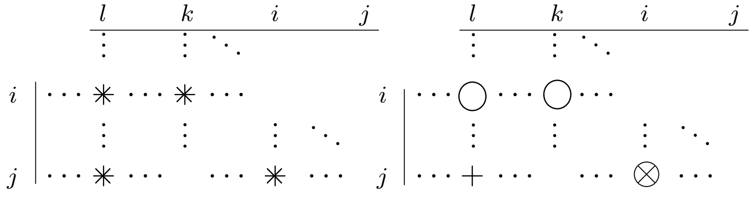
Fig. 6. B (a) and the corresponding binary representation (b) after symbolic elimination.

4 First Ideas on Fill-out

We introduce two different methods for reusing fill-out. The symbols $\otimes$ [$\bigcirc$] represent fill-out that is [not] reused, $*$ represents a non-zero entry and $\times$ represents fill-in in C'.

4.1 Technique 1

We exploit the fact that the elimination of any $c_{j,i}$ yields fill-out in the current memory location. Consider the situation illustrated in Figure 7. The fill-in $c_{j,k}$ can be stored in the memory allocated for $c_{j,i}$ which is getting eliminated. As a consequence the ρ and κ entries corresponding to the fill-in $c_{j,k}$ have to be changed.

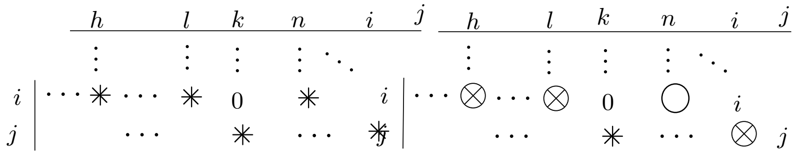


(a) (b)

Fig. 7. (a) before and (b) after the elimination of $c_{j,i}$. The memory allocated for $c_{j,i}$ is reused to store the fill-in $c_{j,k}$.

4.2 Technique 2

Consider in Figure 8 (a) where $j = i + 1$. After the elimination of $c_{j,i}$ the entire row becomes zero. Hence the fill-out in row i can be reused to store the fill-in generated in row j as shown in Figure 8 (b). Consider the CRS of C' in Figure 9 (a) where $c_{6,5}$ has been added to the matrix in Figure 1 (c). Applying above techniques yields 9 fill-out's, 6 fill-in's, and 3 reused fill-out's.



(a) (b)

Fig. 8. Example for the use of second technique : (a) Before and (b) after the elimination of $c_{j,i}$.

The CRS C' after reverse elimination of all intermediate rows has the following format:

$$\begin{aligned}\alpha =&[\mathbf{0},\mathbf{0},\mathbf{0},\mathbf{0},c_{7,1},c_{7,2},c_{7,3},\mathbf{0},\mathbf{0},c_{8,1},c_{8,2},c_{8,3},\mathbf{0},c_{9,1},c_{9,2},c_{9,3},\mathbf{0},\mathbf{0}]\\ \kappa =&[\mathbf{-1},\mathbf{-1},\mathbf{-1},\mathbf{-1},1,2,3,\mathbf{-1},\mathbf{-1},1,2,3,\mathbf{-1},1,2,3,\mathbf{-1},\mathbf{-1}]\\ \rho =&[1,1,1,\mathbf{-1},\mathbf{-1},\mathbf{-1},5,10,14] \quad .\end{aligned}$$

Figure 9 (b) shows the memory access pattern of reverse elimination mapped onto the extended Jacobian.

5 Numerical Results

We compare runtime and memory consumption of our new CRS-based algorithm (**JacAccOnCRS**) with reverse elimination of all intermediate rows of C' (**JacAccOnEJ**). Both methods try to accumulate the Jacobian of the following function:

Listing 1.1. f.cpp

```
void f(double* x, int n, int l) {
  double * h = new double [n];
```

$$\begin{bmatrix}
-1 & & & & & & & & \\
0 & -1 & & & & & & & \\
0 & 0 & -1 & & & & & & \\
c_{4,1} & 0 & c_{4,3} & -1 & & & & & \\
c_{5,1} & c_{5,2} & 0 & 0 & -1 & & & & \\
0 & c_{6,2} & c_{6,3} & 0 & c_{6,5} & -1 & & & \\
0 & 0 & 0 & c_{7,4} & 0 & c_{7,6} & -1 & & \\
0 & 0 & 0 & c_{8,4} & c_{8,5} & 0 & 0 & -1 & \\
0 & 0 & 0 & 0 & c_{9,5} & c_{9,6} & 0 & & -1
\end{bmatrix}
\qquad
\begin{bmatrix}
-1 & & & & & & & & \\
 & -1 & & & & & & & \\
 & & -1 & & & & & & \\
\bigcirc & & \bigcirc & 0 & & & & & \\
\bigcirc & \bigcirc & & & 0 & & & & \\
 & \otimes & \otimes & & & 0 & & & \\
\times & & & \bigcirc & & \bigcirc & -1 & & \\
\times & \times & & \otimes & \bigcirc & & & -1 & \\
\times & \times & \times & & \bigcirc & \bigcirc & & & -1
\end{bmatrix}$$

(a) (b)

Fig. 9. Modified C' (a) and the corresponding memory structure (b) after reverse elimination.

```
for(i=0; i<l; i++){
  if(i%2==0) {
    h[0] = x[n-1]*x[0];
    for(j=1; j<n; j++)
      h[j] = x[j-1]*x[j]; }
  else {
    x[0] = h[n-1]*h[0];
    for(j=1; j<n; j++)
      x[j]=h[j-1]*h[j];
  }
 }
}
```

We set $n = 100$ and $l \in \{10, \cdots, 150\}$. Obviously, $C' \in I\!R^{q \times q}$ where $q = (l+1) \cdot n$. All results have been obtained on an Intel Pentium 4 CPU running at 3.00GHz with 1GB of memory.
JacAccOnCRS performs the following steps:

1. Creation of C' and B by running an overloaded version of F.
2. Fill-in detection by the symbolic algorithm described in Section 2.
3. Allocation CRS and initialization of original non-zero entries.
4. Reverse elimination on C' in CRS format.
5. Extraction of F' from C' in CRS format.

JacAccOnEJ performs the following steps:

1. Creation of C' and B by running an overloaded version of F.
2. Reverse elimination on C'.
3. Extraction of F' from C'.

We observe an overall speedup of roughly five as illustrated in Figure 10. Symbolic reverse elimination on B is about ten times faster than the corresponding procedure on C'. On the given computer architecture we are able to handle problems of sizes $l = 250$ and $l = 1000$ (for $n = 100$) using **JacAccOnEJ** and **JacAccOnCRS**, respectively.

6 Conclusion

Jacobian accumulation on the extended Jacobian can be improved significantly – both in terms of memory requirement and overall runtime – by using static

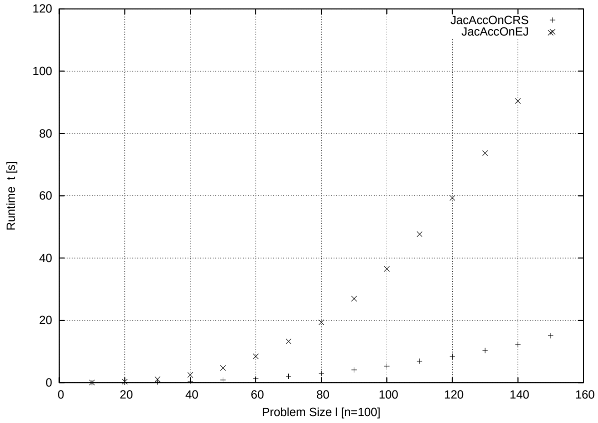


Fig. 10. Runtime of **JacAccOnCRS** vs. **JacAccOnEJ**.

sparse storage allocated based on the result of a symbolic elimination algorithm. The use of fill-out for storing potential fill-in is expected to yield even better results. Further in-depth investigation of the combinatorial and memory access problems is necessary to undercut the runtime of classical Jacobian accumulation techniques such as finite difference approximation or forward mode automatic differentiation [8].

References

1. M. Berz, C. Bischof, G. Corliss, and A. Griewank, editors. *Computational Differentiation: Techniques, Applications, and Tools*, Proceedings Series, Philadelphia, 1996. SIAM.
2. G. Corliss, C. Faure, A. Griewank, L. Hascoet, and U. Naumann, editors. *Automatic Differentiation of Algorithms – From Simulation to Optimization*, New York, 2002. Springer.
3. G. Corliss and A. Griewank, editors. *Automatic Differentiation: Theory, Implementation, and Application*, Proceedings Series, Philadelphia, 1991. SIAM.
4. I. Duff, A. Erisman, and J. Reid. *Direct Methods for Sparse Matrices.* Clarendon Press, Oxford, 1986.
5. A. Griewank. *Evaluating Derivatives. Principles and Techniques of Algorithmic Differentiation.* Number 19 in Frontiers in Applied Mathematics. SIAM, Philadelphia, 2000.
6. A. Griewank and S. Reese. On the calculation of Jacobian matrices by the Markovitz rule. In *[3]*, pages 126–135, 1991.
7. U. Naumann. *Efficient Calculation of Jacobian Matrices by Optimized Application of the Chain Rule to Computational Graphs.* PhD thesis, Technical University of Dresden, December 1999.
8. R. Wengert. A simple automatic derivative evaluation program. *Comm. ACM*, 7:463–464, 1964.

Aachener Informatik-Berichte

This is a list of recent technical reports. To obtain copies of technical reports please consult http://aib.informatik.rwth-aachen.de/ or send your request to: Informatik-Bibliothek, RWTH Aachen, Ahornstr. 55, 52056 Aachen, Email: `biblio@informatik.rwth-aachen.de`

1987-01 * Fachgruppe Informatik: Jahresbericht 1986
1987-02 * David de Frutos Escrig, Klaus Indermark: Equivalence Relations of Non-Deterministic Ianov-Schemes
1987-03 * Manfred Nagl: A Software Development Environment based on Graph Technology
1987-04 * Claus Lewerentz, Manfred Nagl, Bernhard Westfechtel: On Integration Mechanisms within a Graph-Based Software Development Environment
1987-05 * Reinhard Rinn: Über Eingabeanomalien bei verschiedenen Inferenzmodellen
1987-06 * Werner Damm, Gert Döhmen: Specifying Distributed Computer Architectures in AADL*
1987-07 * Gregor Engels, Claus Lewerentz, Wilhelm Schäfer: Graph Grammar Engineering: A Software Specification Method
1987-08 * Manfred Nagl: Set Theoretic Approaches to Graph Grammars
1987-09 * Claus Lewerentz, Andreas Schürr: Experiences with a Database System for Software Documents
1987-10 * Herbert Klaeren, Klaus Indermark: A New Implementation Technique for Recursive Function Definitions
1987-11 * Rita Loogen: Design of a Parallel Programmable Graph Reduction Machine with Distributed Memory
1987-12 J. Börstler, U. Möncke, R. Wilhelm: Table compression for tree automata
1988-01 * Gabriele Esser, Johannes Rückert, Frank Wagner Gesellschaftliche Aspekte der Informatik
1988-02 * Peter Martini, Otto Spaniol: Token-Passing in High-Speed Backbone Networks for Campus-Wide Environments
1988-03 * Thomas Welzel: Simulation of a Multiple Token Ring Backbone
1988-04 * Peter Martini: Performance Comparison for HSLAN Media Access Protocols
1988-05 * Peter Martini: Performance Analysis of Multiple Token Rings
1988-06 * Andreas Mann, Johannes Rückert, Otto Spaniol: Datenfunknetze
1988-07 * Andreas Mann, Johannes Rückert: Packet Radio Networks for Data Exchange
1988-08 * Andreas Mann, Johannes Rückert: Concurrent Slot Assignment Protocol for Packet Radio Networks
1988-09 * W. Kremer, F. Reichert, J. Rückert, A. Mann: Entwurf einer Netzwerktopologie für ein Mobilfunknetz zur Unterstützung des öffentlichen Straßenverkehrs
1988-10 * Kai Jakobs: Towards User-Friendly Networking
1988-11 * Kai Jakobs: The Directory - Evolution of a Standard
1988-12 * Kai Jakobs: Directory Services in Distributed Systems - A Survey
1988-13 * Martine Schümmer: RS-511, a Protocol for the Plant Floor

1988-14 * U. Quernheim: Satellite Communication Protocols - A Performance Comparison Considering On-Board Processing

1988-15 * Peter Martini, Otto Spaniol, Thomas Welzel: File Transfer in High Speed Token Ring Networks: Performance Evaluation by Approximate Analysis and Simulation

1988-16 * Fachgruppe Informatik: Jahresbericht 1987

1988-17 * Wolfgang Thomas: Automata on Infinite Objects

1988-18 * Michael Sonnenschein: On Petri Nets and Data Flow Graphs

1988-19 * Heiko Vogler: Functional Distribution of the Contextual Analysis in Block-Structured Programming Languages: A Case Study of Tree Transducers

1988-20 * Thomas Welzel: Einsatz des Simulationswerkzeuges QNAP2 zur Leistungsbewertung von Kommunikationsprotokollen

1988-21 * Th. Janning, C. Lewerentz: Integrated Project Team Management in a Software Development Environment

1988-22 * Joost Engelfriet, Heiko Vogler: Modular Tree Transducers

1988-23 * Wolfgang Thomas: Automata and Quantifier Hierarchies

1988-24 * Uschi Heuter: Generalized Definite Tree Languages

1989-01 * Fachgruppe Informatik: Jahresbericht 1988

1989-02 * G. Esser, J. Rückert, F. Wagner (Hrsg.): Gesellschaftliche Aspekte der Informatik

1989-03 * Heiko Vogler: Bottom-Up Computation of Primitive Recursive Tree Functions

1989-04 * Andy Schürr: Introduction to PROGRESS, an Attribute Graph Grammar Based Specification Language

1989-05 J. Börstler: Reuse and Software Development - Problems, Solutions, and Bibliography (in German)

1989-06 * Kai Jakobs: OSI - An Appropriate Basis for Group Communication?

1989-07 * Kai Jakobs: ISO's Directory Proposal - Evolution, Current Status and Future Problems

1989-08 * Bernhard Westfechtel: Extension of a Graph Storage for Software Documents with Primitives for Undo/Redo and Revision Control

1989-09 * Peter Martini: High Speed Local Area Networks - A Tutorial

1989-10 * P. Davids, Th. Welzel: Performance Analysis of DQDB Based on Simulation

1989-11 * Manfred Nagl (Ed.): Abstracts of Talks presented at the WG '89 15th International Workshop on Graphtheoretic Concepts in Computer Science

1989-12 * Peter Martini: The DQDB Protocol - Is it Playing the Game?

1989-13 * Martine Schümmer: CNC/DNC Communication with MAP

1989-14 * Martine Schümmer: Local Area Networks for Manufactoring Environments with hard Real-Time Requirements

1989-15 * M. Schümmer, Th. Welzel, P. Martini: Integration of Field Bus and MAP Networks - Hierarchical Communication Systems in Production Environments

1989-16 * G. Vossen, K.-U. Witt: SUXESS: Towards a Sound Unification of Extensions of the Relational Data Model

1989-17 * J. Derissen, P. Hruschka, M.v.d. Beeck, Th. Janning, M. Nagl: Integrating Structured Analysis and Information Modelling

1989-18 A. Maassen: Programming with Higher Order Functions

1989-19 * Mario Rodriguez-Artalejo, Heiko Vogler: A Narrowing Machine for Syntax Directed BABEL

1989-20 H. Kuchen, R. Loogen, J.J. Moreno Navarro, M. Rodriguez Artalejo: Graph-based Implementation of a Functional Logic Language

1990-01 * Fachgruppe Informatik: Jahresbericht 1989

1990-02 * Vera Jansen, Andreas Potthoff, Wolfgang Thomas, Udo Wermuth: A Short Guide to the AMORE System (Computing Automata, MOnoids and Regular Expressions)

1990-03 * Jerzy Skurczynski: On Three Hierarchies of Weak SkS Formulas

1990-04 R. Loogen: Stack-based Implementation of Narrowing

1990-05 H. Kuchen, A. Wagener: Comparison of Dynamic Load Balancing Strategies

1990-06 * Kai Jakobs, Frank Reichert: Directory Services for Mobile Communication

1990-07 * Kai Jakobs: What's Beyond the Interface - OSI Networks to Support Cooperative Work

1990-08 * Kai Jakobs: Directory Names and Schema - An Evaluation

1990-09 * Ulrich Quernheim, Dieter Kreuer: Das CCITT - Signalisierungssystem Nr. 7 auf Satellitenstrecken; Simulation der Zeichengabestrecke

1990-11 H. Kuchen, R. Loogen, J.J. Moreno Navarro, M. Rodriguez Artalejo: Lazy Narrowing in a Graph Machine

1990-12 * Kai Jakobs, Josef Kaltwasser, Frank Reichert, Otto Spaniol: Der Computer fährt mit

1990-13 * Rudolf Mathar, Andreas Mann: Analyzing a Distributed Slot Assignment Protocol by Markov Chains

1990-14 A. Maassen: Compilerentwicklung in Miranda - ein Praktikum in funktionaler Programmierung (written in german)

1990-15 * Manfred Nagl, Andreas Schürr: A Specification Environment for Graph Grammars

1990-16 A. Schürr: PROGRESS: A VHL-Language Based on Graph Grammars

1990-17 * Marita Möller: Ein Ebenenmodell wissensbasierter Konsultationen - Unterstützung für Wissensakquisition und Erklärungsfähigkeit

1990-18 * Eric Kowalewski: Entwurf und Interpretation einer Sprache zur Beschreibung von Konsultationsphasen in Expertensystemen

1990-20 Y. Ortega Mallen, D. de Frutos Escrig: A Complete Proof System for Timed Observations

1990-21 * Manfred Nagl: Modelling of Software Architectures: Importance, Notions, Experiences

1990-22 H. Fassbender, H. Vogler: A Call-by-need Implementation of Syntax Directed Functional Programming

1991-01 Guenther Geiler (ed.), Fachgruppe Informatik: Jahresbericht 1990

1991-03 B. Steffen, A. Ingolfsdottir: Characteristic Formulae for Processes with Divergence

1991-04 M. Portz: A new class of cryptosystems based on interconnection networks

1991-05 H. Kuchen, G. Geiler: Distributed Applicative Arrays

1991-06 * Ludwig Staiger: Kolmogorov Complexity and Hausdorff Dimension

1991-07 * Ludwig Staiger: Syntactic Congruences for w-languages

1991-09 * Eila Kuikka: A Proposal for a Syntax-Directed Text Processing System

1991-10 K. Gladitz, H. Fassbender, H. Vogler: Compiler-based Implementation of Syntax-Directed Functional Programming

1991-11 R. Loogen, St. Winkler: Dynamic Detection of Determinism in Functional Logic Languages

1991-12 * K. Indermark, M. Rodriguez Artalejo (Eds.): Granada Workshop on the Integration of Functional and Logic Programming

1991-13 * Rolf Hager, Wolfgang Kremer: The Adaptive Priority Scheduler: A More Fair Priority Service Discipline

1991-14 * Andreas Fasbender, Wolfgang Kremer: A New Approximation Algorithm for Tandem Networks with Priority Nodes

1991-15 J. Börstler, A. Zündorf: Revisiting extensions to Modula-2 to support reusability

1991-16 J. Börstler, Th. Janning: Bridging the gap between Requirements Analysis and Design

1991-17 A. Zündorf, A. Schürr: Nondeterministic Control Structures for Graph Rewriting Systems

1991-18 * Matthias Jarke, John Mylopoulos, Joachim W. Schmidt, Yannis Vassiliou: DAIDA: An Environment for Evolving Information Systems

1991-19 M. Jeusfeld, M. Jarke: From Relational to Object-Oriented Integrity Simplification

1991-20 G. Hogen, A. Kindler, R. Loogen: Automatic Parallelization of Lazy Functional Programs

1991-21 * Prof. Dr. rer. nat. Otto Spaniol: ODP (Open Distributed Processing): Yet another Viewpoint

1991-22 H. Kuchen, F. Lücking, H. Stoltze: The Topology Description Language TDL

1991-23 S. Graf, B. Steffen: Compositional Minimization of Finite State Systems

1991-24 R. Cleaveland, J. Parrow, B. Steffen: The Concurrency Workbench: A Semantics Based Tool for the Verification of Concurrent Systems

1991-25 * Rudolf Mathar, Jürgen Mattfeldt: Optimal Transmission Ranges for Mobile Communication in Linear Multihop Packet Radio Networks

1991-26 M. Jeusfeld, M. Staudt: Query Optimization in Deductive Object Bases

1991-27 J. Knoop, B. Steffen: The Interprocedural Coincidence Theorem

1991-28 J. Knoop, B. Steffen: Unifying Strength Reduction and Semantic Code Motion

1991-30 T. Margaria: First-Order theories for the verification of complex FSMs

1991-31 B. Steffen: Generating Data Flow Analysis Algorithms from Modal Specifications

1992-01 Stefan Eherer (ed.), Fachgruppe Informatik: Jahresbericht 1991

1992-02 * Bernhard Westfechtel: Basismechanismen zur Datenverwaltung in strukturbezogenen Hypertextsystemen

1992-04 S. A. Smolka, B. Steffen: Priority as Extremal Probability

1992-05 * Matthias Jarke, Carlos Maltzahn, Thomas Rose: Sharing Processes: Team Coordination in Design Repositories

1992-06 O. Burkart, B. Steffen: Model Checking for Context-Free Processes

1992-07 * Matthias Jarke, Klaus Pohl: Information Systems Quality and Quality Information Systems

1992-08 * Rudolf Mathar, Jürgen Mattfeldt: Analyzing Routing Strategy NFP in Multihop Packet Radio Networks on a Line

1992-09 * Alfons Kemper, Guido Moerkotte: Grundlagen objektorientierter Datenbanksysteme

1992-10 Matthias Jarke, Manfred Jeusfeld, Andreas Miethsam, Michael Gocek: Towards a logic-based reconstruction of software configuration management

1992-11 Werner Hans: A Complete Indexing Scheme for WAM-based Abstract Machines

1992-12 W. Hans, R. Loogen, St. Winkler: On the Interaction of Lazy Evaluation and Backtracking

1992-13 * Matthias Jarke, Thomas Rose: Specification Management with CAD

1992-14 Th. Noll, H. Vogler: Top-down Parsing with Simultaneous Evaluation on Noncircular Attribute Grammars

1992-15 A. Schuerr, B. Westfechtel: Graphgrammatiken und Graphersetzungssysteme(written in german)

1992-16 * Graduiertenkolleg Informatik und Technik (Hrsg.): Forschungsprojekte des Graduiertenkollegs Informatik und Technik

1992-17 M. Jarke (ed.): ConceptBase V3.1 User Manual

1992-18 * Clarence A. Ellis, Matthias Jarke (Eds.): Distributed Cooperation in Integrated Information Systems - Proceedings of the Third International Workshop on Intelligent and Cooperative Information Systems

1992-19-00 H. Kuchen, R. Loogen (eds.): Proceedings of the 4th Int. Workshop on the Parallel Implementation of Functional Languages

1992-19-01 G. Hogen, R. Loogen: PASTEL - A Parallel Stack-Based Implementation of Eager Functional Programs with Lazy Data Structures (Extended Abstract)

1992-19-02 H. Kuchen, K. Gladitz: Implementing Bags on a Shared Memory MIMD-Machine

1992-19-03 C. Rathsack, S.B. Scholz: LISA - A Lazy Interpreter for a Full-Fledged Lambda-Calculus

1992-19-04 T.A. Bratvold: Determining Useful Parallelism in Higher Order Functions

1992-19-05 S. Kahrs: Polymorphic Type Checking by Interpretation of Code

1992-19-06 M. Chakravarty, M. Köhler: Equational Constraints, Residuation, and the Parallel JUMP-Machine

1992-19-07 J. Seward: Polymorphic Strictness Analysis using Frontiers (Draft Version)

1992-19-08 D. Gärtner, A. Kimms, W. Kluge: pi-Redˆ+ - A Compiling Graph-Reduction System for a Full Fledged Lambda-Calculus

1992-19-09 D. Howe, G. Burn: Experiments with strict STG code

1992-19-10 J. Glauert: Parallel Implementation of Functional Languages Using Small Processes

1992-19-11 M. Joy, T. Axford: A Parallel Graph Reduction Machine

1992-19-12 A. Bennett, P. Kelly: Simulation of Multicache Parallel Reduction

1992-19-13 K. Langendoen, D.J. Agterkamp: Cache Behaviour of Lazy Functional Programs (Working Paper)

1992-19-14 K. Hammond, S. Peyton Jones: Profiling scheduling strategies on the GRIP parallel reducer

1992-19-15 S. Mintchev: Using Strictness Information in the STG-machine

1992-19-16 D. Rushall: An Attribute Grammar Evaluator in Haskell

1992-19-17 J. Wild, H. Glaser, P. Hartel: Statistics on storage management in a lazy functional language implementation

1992-19-18 W.S. Martins: Parallel Implementations of Functional Languages

1992-19-19 D. Lester: Distributed Garbage Collection of Cyclic Structures (Draft version)

1992-19-20 J.C. Glas, R.F.H. Hofman, W.G. Vree: Parallelization of Branch-and-Bound Algorithms in a Functional Programming Environment

1992-19-21 S. Hwang, D. Rushall: The nu-STG machine: a parallelized Spineless Tagless Graph Reduction Machine in a distributed memory architecture (Draft version)

1992-19-22 G. Burn, D. Le Metayer: Cps-Translation and the Correctness of Optimising Compilers

1992-19-23 S.L. Peyton Jones, P. Wadler: Imperative functional programming (Brief summary)

1992-19-24 W. Damm, F. Liu, Th. Peikenkamp: Evaluation and Parallelization of Functions in Functional + Logic Languages (abstract)

1992-19-25 M. Kesseler: Communication Issues Regarding Parallel Functional Graph Rewriting

1992-19-26 Th. Peikenkamp: Charakterizing and representing neededness in functional loginc languages (abstract)

1992-19-27 H. Doerr: Monitoring with Graph-Grammars as formal operational Models

1992-19-28 J. van Groningen: Some implementation aspects of Concurrent Clean on distributed memory architectures

1992-19-29 G. Ostheimer: Load Bounding for Implicit Parallelism (abstract)

1992-20 H. Kuchen, F.J. Lopez Fraguas, J.J. Moreno Navarro, M. Rodriguez Artalejo: Implementing Disequality in a Lazy Functional Logic Language

1992-21 H. Kuchen, F.J. Lopez Fraguas: Result Directed Computing in a Functional Logic Language

1992-22 H. Kuchen, J.J. Moreno Navarro, M.V. Hermenegildo: Independent AND-Parallel Narrowing

1992-23 T. Margaria, B. Steffen: Distinguishing Formulas for Free

1992-24 K. Pohl: The Three Dimensions of Requirements Engineering

1992-25 * R. Stainov: A Dynamic Configuration Facility for Multimedia Communications

1992-26 * Michael von der Beeck: Integration of Structured Analysis and Timed Statecharts for Real-Time and Concurrency Specification

1992-27 W. Hans, St. Winkler: Aliasing and Groundness Analysis of Logic Programs through Abstract Interpretation and its Safety

1992-28 * Gerhard Steinke, Matthias Jarke: Support for Security Modeling in Information Systems Design

1992-29 B. Schinzel: Warum Frauenforschung in Naturwissenschaft und Technik

1992-30 A. Kemper, G. Moerkotte, K. Peithner: Object-Orientation Axiomatised by Dynamic Logic

1992-32 * Bernd Heinrichs, Kai Jakobs: Timer Handling in High-Performance Transport Systems

1992-33 * B. Heinrichs, K. Jakobs, K. Lenßen, W. Reinhardt, A. Spinner: Euro-Bridge: Communication Services for Multimedia Applications

1992-34 C. Gerlhof, A. Kemper, Ch. Kilger, G. Moerkotte: Partition-Based Clustering in Object Bases: From Theory to Practice

1992-35 J. Börstler: Feature-Oriented Classification and Reuse in IPSEN

1992-36 M. Jarke, J. Bubenko, C. Rolland, A. Sutcliffe, Y. Vassiliou: Theories Underlying Requirements Engineering: An Overview of NATURE at Genesis

1992-37 * K. Pohl, M. Jarke: Quality Information Systems: Repository Support for Evolving Process Models

1992-38 A. Zuendorf: Implementation of the imperative / rule based language PROGRES

1992-39 P. Koch: Intelligentes Backtracking bei der Auswertung funktional-logischer Programme

1992-40 * Rudolf Mathar, Jürgen Mattfeldt: Channel Assignment in Cellular Radio Networks

1992-41 * Gerhard Friedrich, Wolfgang Neidl: Constructive Utility in Model-Based Diagnosis Repair Systems

1992-42 * P. S. Chen, R. Hennicker, M. Jarke: On the Retrieval of Reusable Software Components

1992-43 W. Hans, St.Winkler: Abstract Interpretation of Functional Logic Languages

1992-44 N. Kiesel, A. Schuerr, B. Westfechtel: Design and Evaluation of GRAS, a Graph-Oriented Database System for Engineering Applications

1993-01 * Fachgruppe Informatik: Jahresbericht 1992

1993-02 * Patrick Shicheng Chen: On Inference Rules of Logic-Based Information Retrieval Systems

1993-03 G. Hogen, R. Loogen: A New Stack Technique for the Management of Runtime Structures in Distributed Environments

1993-05 A. Zündorf: A Heuristic for the Subgraph Isomorphism Problem in Executing PROGRES

1993-06 A. Kemper, D. Kossmann: Adaptable Pointer Swizzling Strategies in Object Bases: Design, Realization, and Quantitative Analysis

1993-07 * Graduiertenkolleg Informatik und Technik (Hrsg.): Graduiertenkolleg Informatik und Technik

1993-08 * Matthias Berger: k-Coloring Vertices using a Neural Network with Convergence to Valid Solutions

1993-09 M. Buchheit, M. Jeusfeld, W. Nutt, M. Staudt: Subsumption between Queries to Object-Oriented Databases

1993-10 O. Burkart, B. Steffen: Pushdown Processes: Parallel Composition and Model Checking

1993-11 * R. Große-Wienker, O. Hermanns, D. Menzenbach, A. Pollacks, S. Repetzki, J. Schwartz, K. Sonnenschein, B. Westfechtel: Das SUKITS-Projekt: A-posteriori-Integration heterogener CIM-Anwendungssysteme

1993-12 *	Rudolf Mathar, Jürgen Mattfeldt: On the Distribution of Cumulated Interference Power in Rayleigh Fading Channels
1993-13	O. Maler, L. Staiger: On Syntactic Congruences for omega-languages
1993-14	M. Jarke, St. Eherer, R. Gallersdoerfer, M. Jeusfeld, M. Staudt: ConceptBase - A Deductive Object Base Manager
1993-15	M. Staudt, H.W. Nissen, M.A. Jeusfeld: Query by Class, Rule and Concept
1993-16 *	M. Jarke, K. Pohl, St. Jacobs et al.: Requirements Engineering: An Integrated View of Representation Process and Domain
1993-17 *	M. Jarke, K. Pohl: Establishing Vision in Context: Towards a Model of Requirements Processes
1993-18	W. Hans, H. Kuchen, St. Winkler: Full Indexing for Lazy Narrowing
1993-19	W. Hans, J.J. Ruz, F. Saenz, St. Winkler: A VHDL Specification of a Shared Memory Parallel Machine for Babel
1993-20 *	K. Finke, M. Jarke, P. Szczurko, R. Soltysiak: Quality Management for Expert Systems in Process Control
1993-21	M. Jarke, M.A. Jeusfeld, P. Szczurko: Three Aspects of Intelligent Cooperation in the Quality Cycle
1994-01	Margit Generet, Sven Martin (eds.), Fachgruppe Informatik: Jahresbericht 1993
1994-02	M. Lefering: Development of Incremental Integration Tools Using Formal Specifications
1994-03 *	P. Constantopoulos, M. Jarke, J. Mylopoulos, Y. Vassiliou: The Software Information Base: A Server for Reuse
1994-04 *	Rolf Hager, Rudolf Mathar, Jürgen Mattfeldt: Intelligent Cruise Control and Reliable Communication of Mobile Stations
1994-05 *	Rolf Hager, Peter Hermesmann, Michael Portz: Feasibility of Authentication Procedures within Advanced Transport Telematics
1994-06 *	Claudia Popien, Bernd Meyer, Axel Kuepper: A Formal Approach to Service Import in ODP Trader Federations
1994-07	P. Peters, P. Szczurko: Integrating Models of Quality Management Methods by an Object-Oriented Repository
1994-08 *	Manfred Nagl, Bernhard Westfechtel: A Universal Component for the Administration in Distributed and Integrated Development Environments
1994-09 *	Patrick Horster, Holger Petersen: Signatur- und Authentifikationsverfahren auf der Basis des diskreten Logarithmusproblems
1994-11	A. Schürr: PROGRES, A Visual Language and Environment for PROgramming with Graph REwrite Systems
1994-12	A. Schürr: Specification of Graph Translators with Triple Graph Grammars
1994-13	A. Schürr: Logic Based Programmed Structure Rewriting Systems
1994-14	L. Staiger: Codes, Simplifying Words, and Open Set Condition
1994-15 *	Bernhard Westfechtel: A Graph-Based System for Managing Configurations of Engineering Design Documents
1994-16	P. Klein: Designing Software with Modula-3
1994-17	I. Litovsky, L. Staiger: Finite acceptance of infinite words

1994-18	G. Hogen, R. Loogen: Parallel Functional Implementations: Graphbased vs. Stackbased Reduction
1994-19	M. Jeusfeld, U. Johnen: An Executable Meta Model for Re-Engineering of Database Schemas
1994-20 *	R. Gallersdörfer, M. Jarke, K. Klabunde: Intelligent Networks as a Data Intensive Application (INDIA)
1994-21	M. Mohnen: Proving the Correctness of the Static Link Technique Using Evolving Algebras
1994-22	H. Fernau, L. Staiger: Valuations and Unambiguity of Languages, with Applications to Fractal Geometry
1994-24 *	M. Jarke, K. Pohl, R. Dömges, St. Jacobs, H. W. Nissen: Requirements Information Management: The NATURE Approach
1994-25 *	M. Jarke, K. Pohl, C. Rolland, J.-R. Schmitt: Experience-Based Method Evaluation and Improvement: A Process Modeling Approach
1994-26 *	St. Jacobs, St. Kethers: Improving Communication and Decision Making within Quality Function Deployment
1994-27 *	M. Jarke, H. W. Nissen, K. Pohl: Tool Integration in Evolving Information Systems Environments
1994-28	O. Burkart, D. Caucal, B. Steffen: An Elementary Bisimulation Decision Procedure for Arbitrary Context-Free Processes
1995-01 *	Fachgruppe Informatik: Jahresbericht 1994
1995-02	Andy Schürr, Andreas J. Winter, Albert Zündorf: Graph Grammar Engineering with PROGRES
1995-03	Ludwig Staiger: A Tight Upper Bound on Kolmogorov Complexity by Hausdorff Dimension and Uniformly Optimal Prediction
1995-04	Birgitta König-Ries, Sven Helmer, Guido Moerkotte: An experimental study on the complexity of left-deep join ordering problems for cyclic queries
1995-05	Sophie Cluet, Guido Moerkotte: Efficient Evaluation of Aggregates on Bulk Types
1995-06	Sophie Cluet, Guido Moerkotte: Nested Queries in Object Bases
1995-07	Sophie Cluet, Guido Moerkotte: Query Optimization Techniques Exploiting Class Hierarchies
1995-08	Markus Mohnen: Efficient Compile-Time Garbage Collection for Arbitrary Data Structures
1995-09	Markus Mohnen: Functional Specification of Imperative Programs: An Alternative Point of View of Functional Languages
1995-10	Rainer Gallersdörfer, Matthias Nicola: Improving Performance in Replicated Databases through Relaxed Coherency
1995-11 *	M.Staudt, K.von Thadden: Subsumption Checking in Knowledge Bases
1995-12 *	G.V.Zemanek, H.W.Nissen, H.Hubert, M.Jarke: Requirements Analysis from Multiple Perspectives: Experiences with Conceptual Modeling Technology
1995-13 *	M.Staudt, M.Jarke: Incremental Maintenance of Externally Materialized Views
1995-14 *	P.Peters, P.Szczurko, M.Jeusfeld: Oriented Information Management: Conceptual Models at Work

1995-15 * Matthias Jarke, Sudha Ram (Hrsg.): WITS 95 Proceedings of the 5th Annual Workshop on Information Technologies and Systems
1995-16 * W.Hans, St.Winkler, F.Saenz: Distributed Execution in Functional Logic Programming
1996-01 * Jahresbericht 1995
1996-02 Michael Hanus, Christian Prehofer: Higher-Order Narrowing with Definitional Trees
1996-03 * W.Scheufele, G.Moerkotte: Optimal Ordering of Selections and Joins in Acyclic Queries with Expensive Predicates
1996-04 Klaus Pohl: PRO-ART: Enabling Requirements Pre-Traceability
1996-05 Klaus Pohl: Requirements Engineering: An Overview
1996-06 * M.Jarke, W.Marquardt: Design and Evaluation of Computer–Aided Process Modelling Tools
1996-07 Olaf Chitil: The Sigma-Semantics: A Comprehensive Semantics for Functional Programs
1996-08 * S.Sripada: On Entropy and the Limitations of the Second Law of Thermodynamics
1996-09 Michael Hanus (Ed.): Proceedings of the Poster Session of ALP96 - Fifth International Conference on Algebraic and Logic Programming
1996-09-0 Michael Hanus (Ed.): Proceedings of the Poster Session of ALP 96 - Fifth International Conference on Algebraic and Logic Programming: Introduction and table of contents
1996-09-1 Ilies Alouini: An Implementation of Conditional Concurrent Rewriting on Distributed Memory Machines
1996-09-2 Olivier Danvy, Karoline Malmkjær: On the Idempotence of the CPS Transformation
1996-09-3 Víctor M. Gulías, José L. Freire: Concurrent Programming in Haskell
1996-09-4 Sébastien Limet, Pierre Réty: On Decidability of Unifiability Modulo Rewrite Systems
1996-09-5 Alexandre Tessier: Declarative Debugging in Constraint Logic Programming
1996-10 Reidar Conradi, Bernhard Westfechtel: Version Models for Software Configuration Management
1996-11 * C.Weise, D.Lenzkes: A Fast Decision Algorithm for Timed Refinement
1996-12 * R.Dömges, K.Pohl, M.Jarke, B.Lohmann, W.Marquardt: PRO-ART/CE* — An Environment for Managing the Evolution of Chemical Process Simulation Models
1996-13 * K.Pohl, R.Klamma, K.Weidenhaupt, R.Dömges, P.Haumer, M.Jarke: A Framework for Process-Integrated Tools
1996-14 * R.Gallersdörfer, K.Klabunde, A.Stolz, M.Eßmajor: INDIA — Intelligent Networks as a Data Intensive Application, Final Project Report, June 1996
1996-15 * H.Schimpe, M.Staudt: VAREX: An Environment for Validating and Refining Rule Bases
1996-16 * M.Jarke, M.Gebhardt, S.Jacobs, H.Nissen: Conflict Analysis Across Heterogeneous Viewpoints: Formalization and Visualization
1996-17 Manfred A. Jeusfeld, Tung X. Bui: Decision Support Components on the Internet

1996-18 Manfred A. Jeusfeld, Mike Papazoglou: Information Brokering: Design, Search and Transformation
1996-19 * P.Peters, M.Jarke: Simulating the impact of information flows in networked organizations
1996-20 Matthias Jarke, Peter Peters, Manfred A. Jeusfeld: Model-driven planning and design of cooperative information systems
1996-21 * G.de Michelis, E.Dubois, M.Jarke, F.Matthes, J.Mylopoulos, K.Pohl, J.Schmidt, C.Woo, E.Yu: Cooperative information systems: a manifesto
1996-22 * S.Jacobs, M.Gebhardt, S.Kethers, W.Rzasa: Filling HTML forms simultaneously: CoWeb architecture and functionality
1996-23 * M.Gebhardt, S.Jacobs: Conflict Management in Design
1997-01 Michael Hanus, Frank Zartmann (eds.): Jahresbericht 1996
1997-02 Johannes Faassen: Using full parallel Boltzmann Machines for Optimization
1997-03 Andreas Winter, Andy Schürr: Modules and Updatable Graph Views for PROgrammed Graph REwriting Systems
1997-04 Markus Mohnen, Stefan Tobies: Implementing Context Patterns in the Glasgow Haskell Compiler
1997-05 * S.Gruner: Schemakorrespondenzaxiome unterstützen die paargrammatische Spezifikation inkrementeller Integrationswerkzeuge
1997-06 Matthias Nicola, Matthias Jarke: Design and Evaluation of Wireless Health Care Information Systems in Developing Countries
1997-07 Petra Hofstedt: Taskparallele Skelette für irregulär strukturierte Probleme in deklarativen Sprachen
1997-08 Dorothea Blostein, Andy Schürr: Computing with Graphs and Graph Rewriting
1997-09 Carl-Arndt Krapp, Bernhard Westfechtel: Feedback Handling in Dynamic Task Nets
1997-10 Matthias Nicola, Matthias Jarke: Integrating Replication and Communication in Performance Models of Distributed Databases
1997-11 * R. Klamma, P. Peters, M. Jarke: Workflow Support for Failure Management in Federated Organizations
1997-13 Markus Mohnen: Optimising the Memory Management of Higher-Order Functional Programs
1997-14 Roland Baumann: Client/Server Distribution in a Structure-Oriented Database Management System
1997-15 George Botorog: High-Level Parallel Programming and the Efficient Implementation of Numerical Algorithms
1998-01 * Fachgruppe Informatik: Jahresbericht 1997
1998-02 Stefan Gruner, Manfred Nagel, Andy Schürr: Fine-grained and Structure-Oriented Document Integration Tools are Needed for Development Processes
1998-03 Stefan Gruner: Einige Anmerkungen zur graphgrammatischen Spezifikation von Integrationswerkzeugen nach Westfechtel, Janning, Lefering und Schürr
1998-04 * O. Kubitz: Mobile Robots in Dynamic Environments
1998-05 Martin Leucker, Stephan Tobies: Truth - A Verification Platform for Distributed Systems

1998-06 * Matthias Oliver Berger: DECT in the Factory of the Future
1998-07 M. Arnold, M. Erdmann, M. Glinz, P. Haumer, R. Knoll, B. Paech, K. Pohl, J. Ryser, R. Studer, K. Weidenhaupt: Survey on the Scenario Use in Twelve Selected Industrial Projects
1998-09 * Th. Lehmann: Geometrische Ausrichtung medizinischer Bilder am Beispiel intraoraler Radiographien
1998-10 * M. Nicola, M. Jarke: Performance Modeling of Distributed and Replicated Databases
1998-11 * Ansgar Schleicher, Bernhard Westfechtel, Dirk Jäger: Modeling Dynamic Software Processes in UML
1998-12 * W. Appelt, M. Jarke: Interoperable Tools for Cooperation Support using the World Wide Web
1998-13 Klaus Indermark: Semantik rekursiver Funktionsdefinitionen mit Striktheitsinformation
1999-01 * Jahresbericht 1998
1999-02 * F. Huch: Verifcation of Erlang Programs using Abstract Interpretation and Model Checking — Extended Version
1999-03 * R. Gallersdörfer, M. Jarke, M. Nicola: The ADR Replication Manager
1999-04 María Alpuente, Michael Hanus, Salvador Lucas, Germán Vidal: Specialization of Functional Logic Programs Based on Needed Narrowing
1999-05 * W. Thomas (Ed.): DLT 99 - Developments in Language Theory Fourth International Conference
1999-06 * Kai Jakobs, Klaus-Dieter Kleefeld: Informationssysteme für die angewandte historische Geographie
1999-07 Thomas Wilke: CTL+ is exponentially more succinct than CTL
1999-08 Oliver Matz: Dot-Depth and Monadic Quantifier Alternation over Pictures
2000-01 * Jahresbericht 1999
2000-02 Jens Vöge, Marcin Jurdzinski A Discrete Strategy Improvement Algorithm for Solving Parity Games
2000-03 D. Jäger, A. Schleicher, B. Westfechtel: UPGRADE: A Framework for Building Graph-Based Software Engineering Tools
2000-04 Andreas Becks, Stefan Sklorz, Matthias Jarke: Exploring the Semantic Structure of Technical Document Collections: A Cooperative Systems Approach
2000-05 Mareike Schoop: Cooperative Document Management
2000-06 Mareike Schoop, Christoph Quix (eds.): Proceedings of the Fifth International Workshop on the Language-Action Perspective on Communication Modelling
2000-07 * Markus Mohnen, Pieter Koopman (Eds.): Proceedings of the 12th International Workshop of Functional Languages
2000-08 Thomas Arts, Thomas Noll: Verifying Generic Erlang Client-Server Implementations
2001-01 * Jahresbericht 2000
2001-02 Benedikt Bollig, Martin Leucker: Deciding LTL over Mazurkiewicz Traces
2001-03 Thierry Cachat: The power of one-letter rational languages

2001-04	Benedikt Bollig, Martin Leucker, Michael Weber: Local Parallel Model Checking for the Alternation Free mu-Calculus
2001-05	Benedikt Bollig, Martin Leucker, Thomas Noll: Regular MSC Languages
2001-06	Achim Blumensath: Prefix-Recognisable Graphs and Monadic Second-Order Logic
2001-07	Martin Grohe, Stefan Wöhrle: An Existential Locality Theorem
2001-08	Mareike Schoop, James Taylor (eds.): Proceedings of the Sixth International Workshop on the Language-Action Perspective on Communication Modelling
2001-09	Thomas Arts, Jürgen Giesl: A collection of examples for termination of term rewriting using dependency pairs
2001-10	Achim Blumensath: Axiomatising Tree-interpretable Structures
2001-11	Klaus Indermark, Thomas Noll (eds.): Kolloquium Programmiersprachen und Grundlagen der Programmierung
2002-01 *	Jahresbericht 2001
2002-02	Jürgen Giesl, Aart Middeldorp: Transformation Techniques for Context-Sensitive Rewrite Systems
2002-03	Benedikt Bollig, Martin Leucker, Thomas Noll: Generalised Regular MSC Languages
2002-04	Jürgen Giesl, Aart Middeldorp: Innermost Termination of Context-Sensitive Rewriting
2002-05	Horst Lichter, Thomas von der Maßen, Thomas Weiler: Modelling Requirements and Architectures for Software Product Lines
2002-06	Henry N. Adorna: 3-Party Message Complexity is Better than 2-Party Ones for Proving Lower Bounds on the Size of Minimal Nondeterministic Finite Automata
2002-07	Jörg Dahmen: Invariant Image Object Recognition using Gaussian Mixture Densities
2002-08	Markus Mohnen: An Open Framework for Data-Flow Analysis in Java
2002-09	Markus Mohnen: Interfaces with Default Implementations in Java
2002-10	Martin Leucker: Logics for Mazurkiewicz traces
2002-11	Jürgen Giesl, Hans Zantema: Liveness in Rewriting
2003-01 *	Jahresbericht 2002
2003-02	Jürgen Giesl, René Thiemann: Size-Change Termination for Term Rewriting
2003-03	Jürgen Giesl, Deepak Kapur: Deciding Inductive Validity of Equations
2003-04	Jürgen Giesl, René Thiemann, Peter Schneider-Kamp, Stephan Falke: Improving Dependency Pairs
2003-05	Christof Löding, Philipp Rohde: Solving the Sabotage Game is PSPACE-hard
2003-06	Franz Josef Och: Statistical Machine Translation: From Single-Word Models to Alignment Templates
2003-07	Horst Lichter, Thomas von der Maßen, Alexander Nyßen, Thomas Weiler: Vergleich von Ansätzen zur Feature Modellierung bei der Softwareproduktlinienentwicklung
2003-08	Jürgen Giesl, René Thiemann, Peter Schneider-Kamp, Stephan Falke: Mechanizing Dependency Pairs
2004-01 *	Fachgruppe Informatik: Jahresbericht 2003

2004-02	Benedikt Bollig, Martin Leucker: Message-Passing Automata are expressively equivalent to EMSO logic
2004-03	Delia Kesner, Femke van Raamsdonk, Joe Wells (eds.): HOR 2004 – 2nd International Workshop on Higher-Order Rewriting
2004-04	Slim Abdennadher, Christophe Ringeissen (eds.): RULE 04 – Fifth International Workshop on Rule-Based Programming
2004-05	Herbert Kuchen (ed.): WFLP 04 – 13th International Workshop on Functional and (Constraint) Logic Programming
2004-06	Sergio Antoy, Yoshihito Toyama (eds.): WRS 04 – 4th International Workshop on Reduction Strategies in Rewriting and Programming
2004-07	Michael Codish, Aart Middeldorp (eds.): WST 04 – 7th International Workshop on Termination
2004-08	Klaus Indermark, Thomas Noll: Algebraic Correctness Proofs for Compiling Recursive Function Definitions with Strictness Information
2004-09	Joachim Kneis, Daniel Mölle, Stefan Richter, Peter Rossmanith: Parameterized Power Domination Complexity
2004-10	Zinaida Benenson, Felix C. Gärtner, Dogan Kesdogan: Secure Multi-Party Computation with Security Modules
2005-01 *	Fachgruppe Informatik: Jahresbericht 2004
2005-02	Maximillian Dornseif, Felix C. Gärtner, Thorsten Holz, Martin Mink: An Offensive Approach to Teaching Information Security: “Aachen Summer School Applied IT Security”
2005-03	Jürgen Giesl, René Thiemann, Peter Schneider-Kamp: Proving and Disproving Termination of Higher-Order Functions
2005-04	Daniel Mölle, Stefan Richter, Peter Rossmanith: A Faster Algorithm for the Steiner Tree Problem
2005-05	Fabien Pouget, Thorsten Holz: A Pointillist Approach for Comparing Honeypots
2005-06	Simon Fischer, Berthold Vöcking: Adaptive Routing with Stale Information
2005-07	Felix C. Freiling, Thorsten Holz, Georg Wicherski: Botnet Tracking: Exploring a Root-Cause Methodology to Prevent Distributed Denial-of-Service Attacks
2005-08	Joachim Kneis, Peter Rossmanith: A New Satisfiability Algorithm With Applications To Max-Cut
2005-09	Klaus Kursawe, Felix C. Freiling: Byzantine Fault Tolerance on General Hybrid Adversary Structures
2005-10	Benedikt Bollig: Automata and Logics for Message Sequence Charts
2005-11	Simon Fischer, Berthold Vöcking: A Counterexample to the Fully Mixed Nash Equilibrium Conjecture
2005-12	Neeraj Mittal, Felix Freiling, S. Venkatesan, Lucia Draque Penso: Efficient Reductions for Wait-Free Termination Detection in Faulty Distributed Systems
2005-13	Carole Delporte-Gallet, Hugues Fauconnier, Felix C. Freiling: Revisiting Failure Detection and Consensus in Omission Failure Environments
2005-14	Felix C. Freiling, Sukumar Ghosh: Code Stabilization
2005-15	Uwe Naumann: The Complexity of Derivative Computation

2005-16	Uwe Naumann: Syntax-Directed Derivative Code (Part I: Tangent-Linear Code)
2005-17	Uwe Naumann: Syntax-directed Derivative Code (Part II: Intraprocedural Adjoint Code)
2005-18	Thomas von der Maßen, Klaus Müller, John MacGregor, Eva Geisberger, Jörg Dörr, Frank Houdek, Harbhajan Singh, Holger Wußmann, Hans-Veit Bacher, Barbara Paech: Einsatz von Features im Software-Entwicklungsprozess - Abschlußbericht des GI-Arbeitskreises "Features"
2005-19	Uwe Naumann, Andre Vehreschild: Tangent-Linear Code by Augmented LL-Parsers
2005-20	Felix C. Freiling, Martin Mink: Bericht über den Workshop zur Ausbildung im Bereich IT-Sicherheit Hochschulausbildung, berufliche Weiterbildung, Zertifizierung von Ausbildungsangeboten am 11. und 12. August 2005 in Köln organisiert von RWTH Aachen in Kooperation mit BITKOM, BSI, DLR und Gesellschaft fuer Informatik (GI) e.V.
2005-21	Thomas Noll, Stefan Rieger: Optimization of Straight-Line Code Revisited
2005-22	Felix Freiling, Maurice Herlihy, Lucia Draque Penso: Optimal Randomized Fair Exchange with Secret Shared Coins
2005-23	Heiner Ackermann, Alantha Newman, Heiko Röglin, Berthold Vöcking: Decision Making Based on Approximate and Smoothed Pareto Curves
2005-24	Alexander Becher, Zinaida Benenson, Maximillian Dornseif: Tampering with Motes: Real-World Physical Attacks on Wireless Sensor Networks
2006-03	Uwe Naumann: Intraprocedural Adjoint Code Generated by the Differentiation-Enabled NAGWare Fortran Compiler

[*] These reports are only available as a printed version.

Please contact `biblio@informatik.rwth-aachen.de` to obtain copies.