RWTH Aachen

Department of Computer Science
Technical Report

Divide-and-Color

Joachim Kneis, Daniel Mölle, Stefan Richter, Peter Rossmanith

ISSN 0935–3232 · Aachener Informatik Berichte · AIB-2006-06

RWTH Aachen · Department of Computer Science · May 2006

Divide-and-Color

Joachim Kneis, Daniel Mölle, Stefan Richter, Peter Rossmanith

Lehr- und Forschungsgebiet Theoretische Informatik
RWTH Aachen, Germany
Email: {kneis,moelle,richter,rossmani}@cs.rwth-aachen.de

Abstract. We introduce *divide-and-color*, a new technique for the solution of hard graph problems. It is a combination of the well-known *divide-and-conquer* paradigm and *color-coding* [2]. Our approach first randomly colors all edges or nodes of a graph black and white, and then solves the problem recursively on the two induced parts.
We demonstrate this technique by giving new randomized algorithms for the solution of two important problems. These yield runtime bounds of $O^*(4^k)$ for finding a simple path of length k and $O^*(4^{(h-1)k})$ for finding k edge-disjoint (resp. vertex-disjoint) copies of a graph H with h edges (resp. h nodes) in a given graph. Derandomization gives deterministic algorithms for these problems with running times $O^*(2^{4k})$ and $O^*(2^{4hk})$, respectively.
All these results significantly improve over the currently known best bounds. In particular, our generic algorithms beat specialized ones that have been designed to find k triangles or paths of length two.

1 Introduction

As of today, it is commonly assumed that there are no polynomial-time algorithms for NP-complete problems. On the other hand, these problems occur over and over again in real-life applications, and thus simply need to be dealt with. This contradictory situation has led to the development of several relaxed notions of "solving," such as approximation and randomization. Another approach is called parameterized complexity.

In classical NP-completeness theory, run-time bounds are derived with regard to the worst case taken over all instances of the same size. Informally, the paradigm of parameterized complexity [5] tries to look closer in order to find properties that better measure the hardness of a problem. An instance of length n may still be easy to solve if it fulfills a certain property. If this property can be quantified by a number k—such as the treewidth of a graph, the number of literals that may be set to true in a propositional formula, or the maximum length of a simple path—this number can then be seen as a parameter describing the hardness of an instance.

More precisely, if there exists an algorithm that solves the problem in $O(f(k) \cdot \mathit{poly}(n))$ time for some function f, we say that the problem is fixed-parameter tractable. The corresponding complexity class is called FPT. Since appropriately chosen parameters can be very small in instances occuring in practice, this paradigm often yields rather applicable algorithms. On the other hand, the values of $f(k)$ can be rather large for certain problems. For example, the best parameterized algorithm for Dominating Set Of Queens known today has a running time of $O^*(225^k)$ [3].

There are many techniques for designing FPT-algorithms in the literature, among them kernelization, bounded search trees, dynamic programming, treewidth methods and color-coding. The best runtime bounds for the parameterized versions of some graph problems have been obtained using the latter method, which has been introduced by Alon *et al.* [2]. The underlying idea of this technique is to color the nodes or edges of a graph randomly in order to ease the detection of certain subgraphs. For instance, a path of length k may be hard to find[1], but after coloring the nodes with k colors, a *colorful* path of length $k-1$—i.e., a path that consists of k nodes having k different colors—can be detected quite quickly using dynamic programming.

The runtime bounds obtained via color-coding are usually of the form $O^*(c^k)$ for some constant c. Note that this is only exponential in k and thus much better than, e.g., bounds resulting from Courcelle's famous theorem [4]. The constant c, however, can be very large, as detailed in Section 2. Derandomizing the resulting algorithms [2, 6] leads to bounds of the same form with even larger constants.

In this paper, we present a new and more efficient color-coding technique that can also be derandomized with less dramatic effects on the runtime bound. The crucial idea lies in using only two colors combined with a divide-and-conquer approach. Recursively solving the problem on two subgraphs induced by the coloring turns out to be easier than employing a dynamic programming algorithm as used in classical color-coding.

For a graph G, let $V[G]$ denote the set nodes in G. Similarly, let $E[G]$ denote the set of edges. We apply the new technique to several graph problems, obtaining new runtime bounds of $O^*(2^{2k})$ for LONGEST PATH and $O^*(2^{2(h-1)k})$ for H-GRAPH PACKING, where $h = |V[H]|$, as well as H-GRAPH EDGE-PACKING where $h = |E[H]|$, in the randomized case. The respective bounds for the derandomized case are $O^*(2^{4k})$ and $O^*(2^{4hk})$. Let us first formally define the problems that are to play a rôle in what follows.

Definition 1. *The problem* LONGEST PATH *is defined as follows:*

Input:	*A graph G and a number k*
Parameter:	k
Question:	*Is there a simple path of length k in G?*

The classical color-coding algorithm by Alon *et al.* [2] solves LONGEST PATH in time $O^*((2e)^k) = O^*(2^{2.45k})$ (randomized) or $2^{O(k)}$ with large hidden constants (deterministic). We improve these running times to $O^*(2^{2k})$ and $O^*(2^{4k})$, respectively.

Definition 2. *The problem H-*GRAPH PACKING *is defined as follows:*

Input:	*A graph G and a number k*
Parameter:	k
Question:	*Are there k vertex-disjoint instances of H in G?*

Definition 3. *The problem H-*GRAPH EDGE-PACKING *is defined as follows:*

Input:	*A graph G and a number k*

[1] One way of solving this problem is to look at a depth-first search tree of the graph. If there is a back-edge spanning a path of length k, we are done. Otherwise, there is a tree decomposition of width k, and the problem can be solved using Courcelle's Theorem [4]. Unfortunately, this method yields extremely large values of $f(k)$ even for small k.

Parameter: k
Question: Are there k edge-disjoint instances of H in G?

Until now, the best runtime bounds known for H-Graph Packing and H-Graph Edge-Packing are $O^*(2^{2.45hk})$ (randomized) and $O^*(2^{10hk})$ (deterministic) [6]. We take these to $O^*(2^{2(h-1)k})$ and $O^*(2^{4hk})$, respectively.

Specialized algorithms have been designed for at least two variants of H-Graph Packing and H-Graph Edge-Packing, the first being $K_{2,1}$-Packing [9] and the second Edge-Disjoint Triangle Packing [8]. The corresponding runtime bounds are $O^*(2^{5.301k})$ and $O^*(2^{\frac{9}{2}k\log k+\frac{9}{2}k})$.[2] Our randomized algorithms improve both of these to $O^*(2^{4k})$. Moreover, the derandomized version yields a bound of $O^*(2^{12k})$ for Edge-Disjoint Triangle Packing.

2 Color-Coding

Unfortunately, the papers on color-coding that have been published so far omit a lot of details, especially regarding hidden factors in the respective runtime bounds. This holds for the results on both the problem of finding a path of length k and the problem of finding k occurrences of a fixed subgraph. In the seminal paper on color-coding [2], for example, Alon *et al.* considered k a mere constant. In subsequent papers [6, 8], k was already interpreted as a parameter, but the runtime bounds were stated without giving explicit constants for the basis of the exponential function. This section is dedicated to a more precise analysis of these bounds.

Let us first investigate the algorithm for H-Graph Edge-Packing by Fellows *et al.* [6]: Given a graph G and a number k, does G contain k edge-disjoint occurrences of some fixed subgraph H? Let h denote the number of edges in H. For the analysis, fix a set M of k edge-disjoint occurrences of H in an instance. If coloring all the edges in G randomly using hk colors leaves all the edges in M colored differently, the following algorithm will find these k occurrences of H (or, possibly, some other k occurrences):

For each $S \subseteq \{1, \ldots, hk\}$ where $|S|$ is a multiple of h, use dynamic programming to find out whether G contains $|S|/h$ occurrences of H whose edges have exactly the colors in S. If this is eventually the case for $S = \{1, \ldots, hk\}$, the graph G indeed contains k edge-disjoint occurrences of H. This computation takes $O^*(2^{hk})$ steps.

The probability of coloring all the hk edges in M with different colors is $(hk)!/(hk)^{hk} = \Omega(\sqrt{k}e^{-hk})$. In order to make the failure probability exponentially small, it thus suffices to repeat the entire procedure $O^*(e^{hk})$ times. This gives a total running time of $O^*((2e)^{hk}) = O^*(5.44^{hk})$.

A deterministic algorithm can be obtained via derandomization, as described by Alon *et al.* [2]. It is possible to construct a family $\mathcal{F}$ of hash functions $\{1, \ldots, n\} \to \{1, \ldots, k\}$ such that for every $S \subseteq \{1, \ldots, n\}$ with $|S| = k$ there is one $f \in \mathcal{F}$ that maps S onto $\{1, \ldots, k\}$, i.e., is bijective on S. There are such families of size $2^{O(k)} \log n$ that can be efficiently constructed and evaluated [2].

[2] The latter algorithm is based on kernelization. There also is a $2^{O(k)}$-algorithm based on color-coding. However, the constants hidden in the O-notation are so large that the kernelization-based algorithm is considered more practical.

Instead of coloring the edges randomly, we can consecutively use all members of a family of hash functions $f : E \to \{1, \ldots, hk\}$. This leads to a running time of $2^{O(hk)} n^{O(h)}$.

The constants in the above bound are rather large, but Fellows *et al.* have proposed two ideas on how to improve the running time [6]. The first one is to use more colors in order to increase the probability that all the edges in a solution are assigned different colors. This, however, leads to a more costly dynamic programming phase. The second idea regards the family of hash functions employed: Alon *et al.* did not aim to minimize the size of $\mathcal{F}$ with respect to k. Instead, efficient evaluation was a criterion, but is not needed in this context. For example, a size of $O^*(2^{4hk})$ would be possible at the expense of using $6hk$ colors, leading to an overall running time of only $O^*(2^{10hk})$ [6, 7, 10].

Let us now briefly discuss the original algorithm [2] for Longest Path: We color the nodes randomly using k colors. For a fixed path of length k, there are k^k possible colorings, $k!$ of which make the path *colorful* by assigning k different colors to the k nodes in the path. Hence, the path is colorful with a probability of e^{-k}. A colorful path of length k can be detected in $O^*(2^k)$ steps using dynamic programming. On the other hand, it takes e^k iterations to obtain an exponentially small probability that the path is never colorful. Consequently, this randomized algorithm for Longest Path has a runtime bound of $O^*((2e)^k) = O^*(5.44^k)$.

3 Algorithms based on Divide-and-Color

The basic idea of our new approach is to use only two colors—say, black and white—and to solve a reduced instance of the problem on each of the two induced subgraphs recursively. These two solutions must then be combined into a solution for the original instance. Intuitively, this method has three major advantages:

Firstly, when coloring in stages, a wrong coloring may be amended in the current stage, whereas in classical color-coding, one would have to start from scratch. Secondly, in opposition to classical color-coding, there is no more need for solving the complicated problem addressed by the dynamic programming algorithm. This is because the recursive approach eventually reduces the problem to trivial instances. Thirdly, derandomization becomes a lot easier in the new approach, because we can use almost k-wise independent random variables [1] instead of universal families of hash functions.

3.1 Finding a Path of Length k

Instead of solving the decision problem Longest Path directly, we use an embedding into the following closely related search problem:

Definition 4. *Given a graph $G = (V, E)$, we write $u \xrightarrow{k} v$ for two nodes $u, v \in V$ whenever there is a simple path of length $k - 1$ from u to v.*

The problem Extended Longest Path *is defined as follows:*

Input: *A graph $G = (V, E)$ and a number k*
Parameter: k
Output: *The set $\{ (u, v) \in V^2 \mid u \xrightarrow{k} v \}$.*

We say an algorithm solves EXTENDED LONGEST PATH *with error probability p if for every graph $G = (V, E)$ and every $u, v \in V$ with $u \xrightarrow{k} v$ its result contains (u, v) with probability at least p on input G, k.*

Lemma 1. *Algorithm L from Table 1 solves* EXTENDED LONGEST PATH *with error probability at most $1/4$ in time $O^*(4^k)$.*

Input: A graph $G = (V, E)$ and a number k
Parameter: k
Output: The set $\{ (u, v) \in V^2 \mid u \xrightarrow{k} v \}$.

if $k = 1$ **then return** $\{ (v, v) \mid v \in V \}$ **fi**;
for $3 \cdot 2^k$ times **do**
  Choose some $V' \in 2^V$ with uniform probability;
  $G_1 := G[V']$; $G_2 := G[V - V']$; $R := \emptyset$;
  for all $u, v, w, x \in V$ **do**
    if $(u, v) \in L(G_1, \lceil k/2 \rceil) \wedge \{v, w\} \in E \wedge (w, x) \in L(G_2, \lfloor k/2 \rfloor)$
    then $R := R \cup \{(u, x)\}$ **fi**
  od
od;
return R

Table 1. Algorithm L

Proof. It is easy to see that if Algorithm L returns a pair $\{u, v\}$, then indeed $u \xrightarrow{k} v$. On the other hand, if G contains such a path, L does not return the corresponding pair $\{u, v\}$ with some probability p_k.

Assume there is a simple k-node path from u to v in G. The probability that the first $\lceil k/2 \rceil$ nodes are black and that the other $\lfloor k/2 \rfloor$ nodes are white is 2^{-k}. In that case, $L(G_1)$ and $L(G_2)$ do not contain pairs that allow the algorithm to insert $\{u, v\}$ into R with probability at most $2p_{\lceil k/2 \rceil}$. After $3 \cdot 2^k$ iterations, the probability that $\{u, v\} \notin R$ is at least

$$\left(1 - 2^{-k} + 2^{-k+1} p_{\lfloor k/2 \rfloor}\right)^{3 \cdot 2^k},$$

because with probability $1 - 2^{-k}$ the coloring is bad and with probability at most $2^{-k+1} p_{\lceil k/1 \rceil}$ the coloring is good, but the recursive detection of the black or white subpath fails. We show that $p_k \leq 1/4$ for every k: Obviously $p_1 = 0$, and if $k \geq 2$ then we get $p_k \leq (1 - 2^{-k} + 2^{-k+1}/4)^{3 \cdot 2^k} = (1 - 2^{-(k+1)})^{\frac{3}{2} \cdot 2^{k+1}} \leq e^{-3/2} < 1/4$ by induction on k.

Let T_k denote the number of recursive calls issued by Algorithm L. We obtain the recurrence

$$T_k \leq 3 \cdot 2^k \cdot (T_{\lceil k/2 \rceil} + T_{\lfloor k/2 \rfloor}) \leq 3 \cdot 2^{k+1} T_{\lceil k/2 \rceil}.$$

Using the simple fact that $k + \lceil k/2 \rceil + \lceil \lceil k/2 \rceil / 2 \rceil + \cdots + 1 \leq 2k + \log k$ where the sum ends when a term becomes 1, we get $T_k = O(3^{\log k} 4^k) = O(k^{\log 3} 4^k) = O^*(4^k)$.

Algorithms L thus finds a paths of length k with probability at least $3/4$ if one exists. Iterating it linearly often yields an exponentially small error probability while keeping the running time at $O^*(4^k)$. □

Theorem 1. Longest Path *can be solved with exponentially small error probability in time* $O^*(4^k)$.

Proof. Applying Algorithm L a linear number of times solves the more general problem Extended Longest Path with exponentially small error probability. □

3.2 Graph Packing and Graph Edge-Packing

Input: A graph $G = (V, E)$ and a number k
Parameter: k
Question: Does G contain k edge-disjoint copies of H?

if $k \leq 1$ **then**
  if there are k or more edge-disjoint copies of H **then return** *true*
  else return *false* **fi**
fi;
for $3 \cdot 2^{(h-1)k}$ times **do**
  Choose some $E' \in 2^E$ with uniform probability;
  $G_1 := (V, E')$; $G_2 := (V, E - E')$;
  if $P(G_1, \lceil k/2 \rceil) \wedge P(G_2, \lfloor k/2 \rfloor)$ **then return** *true* **fi**
od;
return *false*

Table 2. Algorithm P

Lemma 2. *Algorithm P from Table 2 solves* H-Graph Edge-Packing *with failure probability* $1/4$ *in time* $O^*(4^{(h-1)k})$, *where* $h = |E[H]|$.

Proof. Assume that H has $h > 1$ edges. In line 6 of the algorithm, E is partitioned into E' and $E - E'$. Let us call the edges in E' *black* and the edges in $E - E'$ *white*. Fix a solution, i.e., a set of k edge-disjoint copies of H in G. For each copy of H, the probability that all its edges have the same color is 2^{-h+1}. The probability that all copies are monochromatic is $2^{-(h-1)k}$ and the probability that exactly $\lceil k/2 \rceil$ copies are black and $\lfloor k/2 \rfloor$ copies are white is at least $2^{-(h-1)k}/k$.

Let p_k be the probability that Algorithm P does not find the fixed solution. Then

$$p_k \leq \left(1 - \frac{2^{-(h-1)k}}{k} + \frac{2^{-(h-1)k}}{k}\left(p_{\lceil k/2 \rceil} + p_{\lfloor k/2 \rfloor}\right)\right)^{3 \cdot 2^{-(h-1)k}}.$$

We show by induction on k that $p_k \leq 1/4$. For $k \leq 1$ we have $p_k = 0$ and if $k > 1$ then

$$p_k \leq \left(1 - \frac{2^{-(h-1)k}}{k} + \frac{2^{-(h-1)k}}{2k}\right)^{3 \cdot 2^{(h-1)k}} \leq \left(1 - 2^{-(h-1)k-1}\right)^{3 \cdot 2^{(h-1)k}} < e^{-3/2}.$$

The runtime analysis is very similar to that of Algorithm L. □

Theorem 2.

1. H-Graph Edge-Packing *can be solved with exponential small error probability in* $O^*(2^{2(h-1)k})$ *steps where* $h = |E[H]|$.

2. *H-Graph Packing can be solved with exponential small error probability in $O^*(2^{2(h-1)k})$ steps where $h = |V[H]|$.*

Proof. For H-Graph Edge-Packing simply repeat the 1/4-error-probability algorithm linearly often in order to amplify the success probability. H-Graph Packing can be solved very similarly. Essentially, we have to color nodes instead of edges. □

Corollary 1. *We can solve* Edge-Disjoint Triangle Packing *in $O^*(2^{4k})$ steps and $K_{1,s}$-*Packing *in $O^*(2^{2sk})$ with exponentially small failure probability.*

4 Derandomization

The common object of the divide-and-color algorithms introduced in the last section lies in finding some subset S of nodes or edges with some property and given size. The first step towards this goal always consists in coloring all nodes or edges black and white. Progress will be made whenever this global coloring implies one particular coloring of S. For example, when looking for a path of length k, we want the first half of this path black, and the second one white. In the randomized approaches detailed above, we color the nodes or edges randomly and repeat this very often, so that the right coloring of S is hit with high probability.

In order to make failure impossible, we have to deterministically cycle through a family of node or edge colorings. Doing so will succeed when we make sure that *every* possible coloring of S is hit at least once. Since we do not know S, however, we need to hit every coloring for any set of size $|S|$ at least once. The notion of (almost) k-wise independence will ease this task for us.

Definition 5. [1] *A set $X \subseteq \{0,1\}^n$ is* k-wise independent *if when $x_1 \dots x_n$ is chosen uniformly from X, then for any k positions $i_1 < \dots < i_k$ and any k-bit string y, we have $\Pr[x_{i_1} \dots x_{i_k} = y] = 2^{-k}$. It is* (ϵ, k)-independent *if $|\Pr[x_{i_1} \dots x_{i_k} = y] - 2^{-k}| \le \epsilon$.*

If we had an $X \subseteq \{0,1\}^n$ that is $(2^{-|S|-1}, |S|)$-independent, we could color a set of n nodes or edges according to each element in X with the guarentee that S is colored in every possible way at least once. Fortunately, Alon *et al.* have found a construction that does the job.

Proposition 1. [1] *Let $n = 2^t - 1$ and let k be an odd integer. Then it is possible to construct n bits that are (ϵ, k)-independent using $2\left(\lceil \log \frac{1}{\epsilon} + \log(1+(k-1)t/2) \rceil\right)$ bits.*

Moreover, their construction works in quasi-linear time. Notice that the number of bits in the seed of an $(2^{-|S|-1}, |S|)$-wise independent set X can be bounded by $2|S| + 2\log|S| + 2\log\log n + 2$. The cardinality of X is thus no more than $O(4^{|S|} \cdot |S|^2 \cdot \log^2 n)$ and X can be constructed in $O^*(4^{|S|})$ time.

Theorem 3. *Algorithm P' solves H-Graph Edge-Packing in time $O^*(16^{hk})$, where $h = |E[H]|$.*

Input: A graph $G = (V, E)$, $E = \{e_1, \ldots, e_m\}$ and a number k
Parameter: k
Question: Does G contain k edge-disjoint copies of H?

```
if k ≤ 1 then
  if there are k or more edge-disjoint copies of H then return true
  else return false fi
fi;
for all s ∈ {0,1}^⌈2hk+2 log(hk)+2 log log |E|⌉+2 do
  Expand s into x ∈ {0,1}^|E| as in Proposition 1;
  E' := { e_i ∈ E | x_i = 1 };
  G_1 := (V, E'); G_2 := (V, E − E');
  if P(G_1, ⌈k/2⌉) ∧ P(G_2, ⌊k/2⌋) then return true fi
od;
return false
```

Table 3. Algorithm P′

Proof. There are $O(4^{hk}(hk)^2 \log^2 |E|) = O^*(4^{hk})$ possibilities to choose s. Since the algorithm calls itself recursively, the total number of recursive calls is $O^*(16^{hk})$.

The correctness follows from the fact that for a solution consisting of k copies of H with a total of hk edges, E' eventually contains exactly the edges of $\lceil k/2 \rceil$ many of those copies. □

Analogously, we can prove the following theorem.

Theorem 4. *There is a deterministic algorithm that solves* Longest Path *in time* $O^*(16^k)$.

5 Conclusion

We have introduced a new technique and applied it to design faster algorithms for several problems. However, we did not exhaust the method's full potential. In particular, we think that the cost incurred by derandomization could be decreased. While almost k-wise independent random variables seem to be the right tool for derandomizing the Longest Path algorithm, a weaker property suffices for packing problems. In the case of Triangle Packing, for instance, we used almost $3k$-independent random variables to make sure we encounter every possible coloring of the $3k$ edges of k triangles. It would be enough, however, if for every possible way of grouping $3k$ edges into k triangles, exactly half of these triangles were colored black and the others white at least once. Designing appropriate sample spaces thus seems to be a promising goal.

Acknowledgement. We thank Martin Dietzfelbinger for pointing us in the direction of almost k-wise independent random variables.

References

1. N. Alon, O. Goldreich, J. Håstad, and R. Peralta. Simple constructions of almost k-wise independent random variables. *Journal of Random structures and Algorithms*, 3(3):289–304, 1992.
2. N. Alon, R. Yuster, and U. Zwick. Color-coding. *J. ACM*, 42(4):844–856, 1995.

3. M. Cesati. Compendium of parameterized problems, 2004. `http://bravo.ce.uniroma2.it/home/cesati/research/compendium.ps`.
4. B. Courcelle, J. A. Makowsky, and U. Rotics. Linear time solvable optimization problems on graphs of bounded clique width. In *Proc. of 24th WG*, number 1517 in LNCS, pages 1–16. Springer, 1998.
5. R. G. Downey and M. R. Fellows. *Parameterized Complexity*. Springer-Verlag, 1999.
6. M. R. Fellows et al. Faster fixed-parameter tractable algorithms for matching and packing problems. In *Proc. of 12th ESA*, number 3221 in LNCS, pages 311–322. Springer, 2004.
7. M. Fredman, J. Komlos, and E. Szemeredi. Storing a sparse table with $O(1)$ worst case access time. In *Proc. of 23d FOCS*, pages 165–169, 1982.
8. L. Mathieson, E. Prieto, and P. Shaw. Packing edge disjoint triangles: A parameterized view. In *Proc. of 1st IWPEC*, number 3162 in LNCS, pages 127–137. Springer, 2004.
9. E. Prieto and C. Sloper. Looking at the stars. In *Proc. of 1st IWPEC*, number 3162 in LNCS. Springer, 2004.
10. C. F. Slot and P. van Emde Boas. On tape versus core; an application of space efficient hash functions to the invariance of space. *Elektronische Informationsverarbeitung und Kybernetik*, 21(4/5):246–253, 1985.

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-Red^+ - 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

2006-04 Ebadollah Varnik, Uwe Naumann, Andrew Lyons: Toward Low Static Memory Jacobian Accumulution

* These reports are only available as a printed version.

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