Covers have structure

Henning Wunderlich

Ulmer Informatik-Berichte

Nr. 2008-08
Juli 2008

Covers have structure

Henning Wunderlich *

July 8, 2008

Abstract

We establish a new connection between the theory of Berge graphs (perfect graphs) and communication complexity. We discover a new class of square-free Berge graphs, the class of *beautiful graphs*, and make progress towards their characterization: on the one hand, we give a complete list of forbidden induced subgraphs of order ≤ 7, on the other hand, we show that every square-free bipartite graph is beautiful, and, as the main result, we characterize the beautiful line graphs of square-free bipartite graphs.

1 Introduction

1.1 Theory of perfect graphs

Shannon [11, 12] considered zero-error data transmission and reduced the problem of determining the zero-error channel capacity to a problem in graph theory, namely calculating $\sup_{n\to\infty} \frac{1}{n} \log \omega(G^n)$ (now called *Shannon zero-error capacity*), where G is a graph associated with the given channel, G^n is its n-th graph power, and $\omega(G)$ is the clique number of G. The n-th graph power G^n is the strong graph product of n copies of g; given graphs G_1 and G_2 the strong graph product is a graph with vertex set $V(G_1) \times V(G_2)$ and two distinct vertices are connected iff they are adjacent or equal in each coordinate. Determining the Shannon zero-error capacity is extremely hard in general, e.g. see [1, 8], but easily solved for so called *perfect graphs*, introduced by Berge [2]. These are graphs for which the chromatic and clique number have the same value for each induced subgraph. Berge conjectured that a graph is perfect iff it does not contain any odd holes or odd antiholes. An induced cycle of odd length at least 5 is called an *odd hole*, while an induced subgraph that is the complement of an odd hole is called an *odd antihole*. Graphs without odd holes and odd antiholes are called *Berge graphs*. The above conjecture was known as the *Strong Perfect Graph Conjecture*, which, based on a series of works, especially [4], was finally answered in the affirmative by Chudnovsky, Robertson, Seymour and Thomas [3] in May, 2002.
In the sequel we will also need the following notions: A 4-cycle is called a *square*; a *square-free* graph does not contain a square as an induced subgraph. The *line graph* of the graph G is the graph $L(G)$ whose nodes are the edges of G and two nodes u, v of $L(G)$ are adjacent in $L(G)$ iff the edges u, v of G are incident

*Universität Ulm, Fakultät für Ingenieurwissenschaften und Informatik, Institut für Theoretische Informatik, Oberer Eselsberg, D-89069 Ulm, e-mail: Henning.Wunderlich@uni-ulm.de

to a common node of G. We write $G =_{iso} H$ iff G and H are isomorphic, and $G \leq_{iso} H$ iff G is isomorphic to an induced subgraph of H. For introductions to graph theory and the theory of perfect graphs, we refer the reader to [5, 10].

1.2 Basics from communication complexity

In 1979 Yao [13] introduced a two player communication model: Let $\mathcal{X}, \mathcal{Y}, \mathcal{Z}$ be finite sets. Player Alice has input $x \in \mathcal{X}$, player Bob has $y \in \mathcal{Y}$. Both want to compute $f(x, y)$ for a function $f \colon \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$, or they want to compute a relation, i.e. a value $z \in \mathcal{Z}$ such that $(x, y, z) \in R$ for a relation $R \subseteq \mathcal{X} \times \mathcal{Y} \times \mathcal{Z}$. The communication between the two players is specified by protocols. We will not delve into definitions of protocols and communication complexity. For an excellent introduction to this subject we refer to [7]. Important for us is that a (nondeterministic) protocol induces a cover of the communication matrix of the function/relation via monochromatic combinatorial rectangles. These notions are defined below.

A matrix $M \colon \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ is called a *function matrix over* $\mathcal{X}, \mathcal{Y}, \mathcal{Z}$, and a matrix $M \colon \mathcal{X} \times \mathcal{Y} \to \mathcal{P}(\mathcal{Z})$ is called a *relation matrix over* $\mathcal{X}, \mathcal{Y}, \mathcal{Z}$ iff for all rows $x \in \mathcal{X}$ and columns $y \in \mathcal{Y}$ we have $M_{x,y} \neq \emptyset$. A *combinatorial rectangle in* M is a set $R = A \times B$, $A \subseteq \mathcal{X}$, $B \subseteq \mathcal{Y}$. For $R = C \times D$ we define $A(R) := C$ and $B(R) := D$. Let $z \in \mathcal{Z}$. If M is a function matrix, a combinatorial rectangle $R = A \times B$ is called *z-chromatic in M* iff for all $x \in A$, $y \in B$ we have $M_{x,y} = z$. If M is a relation matrix, R is called *z-chromatic in M* iff for all $x \in A$, $y \in B$ we have $z \in M_{x,y}$. R is called *monochromatic* iff there exists $z \in \mathcal{Z}$ such that R is z-chromatic.

Definition 1.1. Let M be a function or relation matrix over $\mathcal{X}, \mathcal{Y}, \mathcal{Z}$.

- A combinatorial rectangle R is *nonextendible* iff R is monochromatic in M and adding rows or columns to R results in a nonmonochromatic rectangle.
- We associate with M its *cover structure graph* $\mathcal{G}(M) := (V(M), E(M))$ (cs-graph for short),

$$\begin{aligned} V(M) &:= \{R \mid R \text{ nonextendible rectangle in } M\} \\ E(M) &:= \{\{R, R'\} \mid R, R' \in V(M), R \neq R', R \cap R' \neq \emptyset\} \end{aligned}$$

- Let $z \in \mathcal{Z}$. We also associate with M its *z-chromatic cover structure graph* $\mathcal{G}^z(M) := (V^z(M), E^z(M))$,

$$\begin{aligned} V^z(M) &:= \{R \mid R \text{ nonextendible } z\text{-chromatic rectangle in } M\} \\ E^z(M) &:= \{\{R, R'\} \mid R, R' \in V^z(M), R \neq R', R \cap R' \neq \emptyset\} \end{aligned}$$

1.3 Communication complexity and Berge graphs

As we mentioned earlier, Berge graphs play an important role in noninteractive communication complexity, i.e. information theory, in the context of zero-error data transmission and the determination of channel capacities. But no connection was known before in the interactive case, i.e. communication complexity. We show, that for total functions f the covers of their communication matrices M_f *have structure* in the sense, that their cover structure graphs $\mathcal{G}(M_f)$ are not arbitrary. For an important subclass of the cover structure graphs, the *beautiful graphs*, we prove that this class is strictly contained in the class of square-free Berge graphs, thus establishing a connection between Berge graphs and interactive communication.

The following result might lead to the conclusion, that cs-graphs are uninteresting. However, for function matrices the situation is completely different, as we will see in Theorem 2.3. We denote with $[n]$ the set $\{1, \dots, n\}$ of the first n natural numbers.

Theorem 1.2. *Let G be an arbitrary graph. Then there exists a* relation *matrix M, such that $G =_{iso} \mathcal{G}(M)$.*

Proof. W.l.o.g. assume $G = (V, E)$, $V = [n]$. We define the $1 \times n^2$-block matrix M with values in $\mathcal{P}([n])$ by $M := (B^{(1)}, \dots, B^{(n)})$, where each block $B^{(i)}$ is a $1 \times n$-matrix defined by $B_j^{(i)} := \{i, j\}$, if $\{i, j\} \in E$, and $B_j^{(i)} := \{i\}$ otherwise. For each color $i \in [n]$ there exists exactly one nonextendible rectangle $R_i := \{1\} \times \{j \mid i \in M_{1,j}\}$. Thus, $V(M) = \{R_i \mid i \in [n]\}$. If $\{i, j\} \in E$, then R_i, R_j intersect in block position $B_j^{(i)}$ (and $B_i^{(j)}$) implying $\{R_i, R_j\} \in E(M)$. Conversely, if $\{R_i, R_j\} \in E(M)$, then there exist $k, l \in [n]$, such that R_i, R_j intersect in $B_l^{(k)}$. The case $k \notin \{i, j\}$ cannot occur by construction ($|B_j^{(i)}| \leq 2$). W.l.o.g. assume $k = i$. Necessarily, $B_l^{(i)} = \{i, j\}$. Thus, $l = j$ and $\{i, j\} \in E$. We conclude $E(M) = \{\{R_i, R_j\} \mid \{i, j\} \in E\}$ proving $G =_{iso} \mathcal{G}(M)$. $\square$

Given $z \in \mathcal{Z}$ and a function matrix M over $\mathcal{X}, \mathcal{Y}, \mathcal{Z}$, define the corresponding $\{0, 1\}$-valued matrix $M^{(z)}$ by $M_{x,y}^{(z)} := 1$, if $M_{x,y} = z$, and $M_{x,y}^{(z)} := 0$ otherwise. As rectangles with different colors do not intersect for function matrices, we get $\mathcal{G}(M) =_{iso} \bigcup_{z \in \mathcal{Z}} \mathcal{G}^z(M^{(z)})$, where $\bigcup$ denotes the disjoint union of graphs. Thus, we only need to deal with cs-graphs of function matrices over finite sets $\mathcal{X}, \mathcal{Y}$ and $\mathcal{Z} = \{0, 1\}$. From here on, when we talk about *matrices*, we mean function matrices over finite sets $\mathcal{X}, \mathcal{Y}$ and $\mathcal{Z} = \{0, 1\}$. We also write $\mathcal{G}(M)$, when we mean $\mathcal{G}^1(M)$. We call matrices M with $G = \mathcal{G}(M)$ *representations* of G. We denote the class of *cover structure graphs (cs-graphs)*, i.e. the class of graphs which can be represented by function matrices, with **csg**.

1.4 Easy observations concerning cs-graphs

In this subsection we prove several easy results about cs-graphs and state structural properties. The independent set $\overline{K}_n$, the complete graph K_n and even cycles C_{2n} are cs-graphs, $n \in \mathbb{N}$, as can be seen by looking at the identity matrix E_n and the matrices over $\mathcal{X} = \mathcal{Y} = [n]$ defined below: For $m, n \in \mathbb{N}$, $m \leq n$, define the following $n \times n$-matrix by $({}_{\text{rep}}\mathrm{K}_m^{(n)})_{i,j} := 1$, if $i \leq m + 1 - j$, and $({}_{\text{rep}}\mathrm{K}_m^{(n)})_{i,j} := 0$ otherwise. Then $K_m =_{iso} \mathcal{G}({}_{\text{rep}}\mathrm{K}_m^{(n)})$, and $\overline{K}_n =_{iso} \mathcal{G}(E_n)$. The matrix ${}_{\text{rep}}\mathrm{K}_m^{(n)}$ as a representation for K_m is defined more general than is needed here, because we need it later in this form. A possible representation for the even cycle C_{2n} is the following $n \times n$-matrix:

$$
{}_{\mathrm{rep}}\mathrm{C}_{2n} \quad := \quad \begin{pmatrix} 1 & 1 & 0 & \cdots & \cdots & 0 \\ 0 & 1 & 1 & & & \vdots \\ \vdots & & 1 & 1 & & \vdots \\ \vdots & & & & \ddots & 0 \\ 0 & & & & 1 & 1 \\ 1 & 0 & \cdots & \cdots & 0 & 1 \end{pmatrix}
$$

Clearly, we have $C_{2n} =_{iso} \mathcal{G}({}_{\mathrm{rep}}\mathrm{C}_{2n})$.

In the sequel we only consider connected cs-graphs. On the one hand, if G has connected components G_1 and G_2 represented by M_1 and M_2, respectively, then the block diagonal matrix $M := \mathrm{diag}(M_1, M_2)$ is a representation of G. On the other hand, one can show that if G is a cs-graph, then its components G_1 and G_2 are also cs-graphs: If in a representation M of G a rectangle R_1 representing a node $v_1 \in G_1$ would share a row or column with a rectangle R_2 representing a node $v_2 \in G_2$, then there would exist a nonextendible rectangle J incident to both G_1 and G_2. As M is a representation of G the rectangle J would represent a node adjacent to both v_1 and v_2 in contradiction to the assumption that G_1 and G_2 are different connected components. If A_i, B_i denote the rows and columns covered by rectangles representing nodes in G_i, then $A_1 \cap A_2, B_1 \cap B_2 = \emptyset$. Thus, a permutation of the rows and columns of M yields a representation $diag(M_1, M_2)$ of G.

2 Covers have structure

In this section we show that in contrast to the case of relation matrices not every graph is a cs-graph of a function matrix. Thus, in contrast to relations for total functions the corresponding cs-graphs are not arbitrary implying that (for total functions) covers have structure. This might explain why there are phenomenological differences (see e.g. [7, Chap. 5]) in the communication complexity of relations compared to the communication complexity of total functions.

An important observation is that nonextendible combinatorial rectangles cannot intersect in an arbitrary fashion. Only two modes of intersection are possible, namely *cross* and *spade* situations (see Figure 1).

Definition 2.1. Let M be a matrix, and let $R_i := A_i \times B_i \in V(M)$, $i \in [2]$, $\{R_1, R_2\} \in E(M)$.

- If $A_1 \subsetneq A_2$ and $B_2 \subsetneq B_1$, then we have a *cross situation* $cross(R_1, R_2)$.
- If $A_1 - A_2, A_2 - A_1, B_1 - B_2, B_2 - B_1 \neq \emptyset$, then we have a *spade situation* $spade\{R_1, R_2\}$.

Note that while $spade\{R_1, R_2\}$ implies $spade\{R_2, R_1\}$ in case $cross(R_1, R_2)$ the situation $cross(R_2, R_1)$ does not occur. In case we do not care which cross situation holds, we let $cross\{R_1, R_2\} := cross(R_1, R_2) \vee cross(R_2, R_1)$ denote the symmetrized version.

In the following lemma we list helpful observations we will extensively use in the sequel.

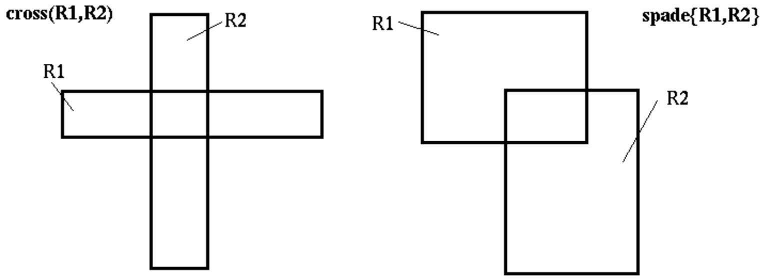

Figure 1: cross and spade situations

Lemma 2.2. *Let M be a matrix, and let $R_i := A_i \times B_i \in V(M)$, $i \in [3]$.*

1. *If $\{R_1, R_2\} \in E(M)$, then exactly one of the following situations occurs:*
$$cross(R_1, R_2), \quad cross(R_2, R_1), \quad spade\{R_1, R_2\}.$$
2. *If we have $cross(R_1, R_2)$ and $cross(R_2, R_3)$, then $\{R_1, R_3\} \in E(M)$.*
3. *Let M be a matrix, and let $R_i := A_i \times B_i \in V(M)$, $i \in [3]$. If we have $spade\{R_1, R_2\}$, then $K_4 \leq_{iso} \mathcal{G}(M)$.*
4. *We assume $cross(R_1, R_2)$, $cross(R_3, R_2)$ and $\{R_1, R_3\} \notin E(M)$. If $B_2 \subsetneq B_1 \cap B_3$, then there exists $R_4 \in V(M)$, such that $\{R_i, R_4\} \in E(M)$ for all $i \in [3]$.*

Proof. 1. By case distinction: *Case* $A_1 = A_2$. Here $R_1 = R_2$, or at least one of R_1, R_2 is extendible, a contradiction. *Case* $A_1 \subsetneq A_2$. If $B_1 \subsetneq B_2$ or $B_1 - B_2, B_2 - B_1 \neq \emptyset$ then R_1 is extendible, a contradiction. If $B_2 \subsetneq B_1$ then we have $cross(R_1, R_2)$. *Case* $A_2 \subsetneq A_1$. Analogous to $A_1 \subsetneq A_2$. *Case* $A_1 - A_2, A_2 - A_1 \neq \emptyset$. All cases are analogous to the previous ones, except $B_1 - B_2, B_2 - B_1 \neq \emptyset$, where we have a spade situation $spade\{R_1, R_2\}$.

2. Both R_1, R_3 cover $A_1 \times B_3$ by nonextendibility, and thus intersect. From $cross(R_1, R_2)$ and $cross(R_2, R_3)$ it also follows $A_1 \subsetneq A_2$ and $A_2 \subsetneq A_3$, respectively. Thus, $A_1 \subsetneq A_3$, which implies $R_1 \neq R_3$.

3. Let R_3, R_4 be arbitrary nonextendible combinatorial rectangles in M covering $(A_1 \cap A_2) \times (B_1 \cup B_2)$ and $(A_1 \cup A_2) \times (B_1 \cap B_2)$, respectively. Clearly, $R_1, \ldots, R_4$ are pairwise distinct. As all of them cover $(A_1 \cap A_2) \times (B_1 \cap B_2)$ they pairwise intersect. Thus, $\mathcal{G}(M)(\{R_1, \ldots, R_4\}) =_{iso} K_4$.

4. Let $R_4 \in V(M)$ be an arbitrary nonextendible combinatorial rectangle covering $(A_1 \cup A_3) \times (B_1 \cap B_3)$. From $cross(R_1, R_2)$ it follows $A_1 \subsetneq A_2$. As $B_2 \subsetneq B_1 \cap B_3$, we get $R_4 \cap R_2 \neq \emptyset$ and $R_4 \neq R_2$. By construction we also have $R_4 \cap R_1, R_4 \cap R_3 \neq \emptyset$. From $\{R_1, R_3\} \notin E(M)$ and $\emptyset \neq B_2 \subsetneq B_1 \cap B_3$ we derive $A_1 \cap A_3 = \emptyset$. Thus, $R_4 \neq R_1$ and $R_4 \neq R_3$. $\square$

Now we can show that not all graphs are cs-graphs:

Theorem 2.3. *The square C_4, odd holes C_{2n+1}, $n \geq 2$, and the graphs gem, star*[1] *and watch (see Figure 2) are not cs-graphs.*

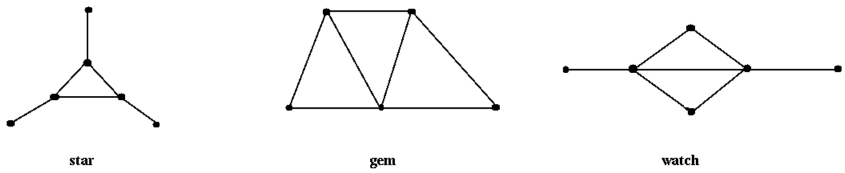

Figure 2: Star, gem and watch

Proof. Due to the many case distinctions we recommend that the reader visualizes the proofs by drawing the cross situations under consideration (we cannot do this here for space reasons).

1. We assume that C_4 is a cs-graph. Then there exists a matrix M such that $C_4 =_{iso} \mathcal{G}(M)$. We have $V(M) = \{R_1, \ldots, R_4\}$ and $E(M) = \{\{R_1, R_2\}, \{R_2, R_3\}, \{R_3, R_4\}, \{R_4, R_1\}\}$. By Lemma 2.2 (1, 3) for R_i, R_{i+1} and R_4, R_1 only cross situations are possible, as $K_4 \not\leq_{iso} C_4$. W.l.o.g. we assume $cross(R_1, R_2)$. Then by Lemma 2.2 (2) we must have $cross(R_3, R_2)$, as $C_3 \not\leq_{iso} C_4$. Applying Lemma 2.2 (4) yields $B_1 \cap B_3 = B_2$. An analogous argumentation (consider the transpose of M) for R_2, R_3, R_4 yields $A_2 \cap A_4 = A_3$. From $R_1 \cap R_3 = \emptyset$ and $B_1 \cap B_3 = B_2 \neq \emptyset$ it follows $A_1 \cap A_3 = \emptyset$. Then we have $A_4 = A_3 \cup (A_4 - A_2)$, and thus $A_4 \cap A_1 = (A_3 \cap A_1) \cup ((A_4 - A_2) \cap A_1) = \emptyset \cup \emptyset = \emptyset$ using $A_1 \subseteq A_2$. But this implies $R_1 \cap R_4 = \emptyset$ contradicting $\{R_1, R_4\} \in E(M)$. We conclude that C_4 cannot be a cs-graph.

2. We assume that C_{2n+1} is a cs-graph for $n \geq 2$. Then there exists a matrix M, such that $C_{2n+1} =_{iso} \mathcal{G}(M)$. We have $V(M) = \{R_1, \ldots, R_{2n+1}\}$ and $E(M) = \{\{R_i, R_{i+1}\} \mid i \in [2n]\} \cup \{\{R_{2n+1}, R_1\}\}$. As $K_4 \not\leq_{iso} C_{2n+1}$, by Lemma 2.2 (1, 3) only cross situations are possible. W.l.o.g. we assume $cross(R_1, R_2)$. As $C_3 \not\leq_{iso} C_{2n+1}$ iteratively applying Lemma 2.2 (2) yields the sequence $cross(R_3, R_2)$, $cross(R_3, R_4)$, $\ldots$, $cross(R_{2n+1}, R_{2n})$, and thus $cross(R_{2n+1}, R_1)$. But going backwards starting from $cross(R_1, R_2)$ gives us $cross(R_1, R_{2n+1})$. We get $cross(R_{2n+1}, R_1)$ and $cross(R_1, R_{2n+1})$, a contradiction. We conclude that C_{2n+1} cannot be a cs-graph.

3. We assume that gem is a cs-graph, i.e. gem $=_{iso} \mathcal{G}(M)$ for a matrix M. We have $V(M) = \{R_1, \ldots, R_5\}$ and $E(M) = \{\{R_1, R_2\}, \{R_1, R_5\}, \{R_2, R_3\}, \{R_2, R_4\}, \{R_2, R_5\}, \{R_3, R_4\}, \{R_4, R_5\}\}$. As $K_4 \not\leq_{iso}$ gem, only cross situations are possible. W.l.o.g. we assume $cross(R_1, R_2)$. $\{R_1, R_3\}, \{R_1, R_4\} \notin E(M)$ implies $cross(R_3, R_2)$ and $cross(R_4, R_2)$, respectively. *Case 1:* Assume $cross(R_3, R_4)$. *Case 1.1:* Assume $cross(R_1, R_5)$. $cross(R_2, R_5)$ implies $cross(R_3, R_5)$ contradicting $R_3 \cap R_5 = \emptyset$. Thus, assume $cross(R_5, R_2)$. *Case 1.1.1:* Assume $cross(R_4, R_5)$. Then $A(R_3) \subseteq A(R_4) \subseteq A(R_5)$ and $B(R_5) \subseteq B(R_4) \subseteq B(R_3)$.

[1] The star graph is also called *net* in many publications.

But $R_3 \cap R_5 = (A(R_3) \cap A(R_5)) \times (B(R_3) \cap B(R_5)) \supseteq A(R_3) \times B(R_5) \neq \emptyset$, a contradiction. *Case 1.1.2:* Assume $cross(R_5, R_4)$. We must have $A(R_1) \cap A(R_4) = \emptyset$, as $B(R_1) \cap B(R_4) \supseteq B(R_2) \neq \emptyset$ and $R_1 \cap R_4 = \emptyset$. But then $A(R_5) \subseteq A(R_4)$ implies $A(R_1) \cap A(R_5) = \emptyset$ contradicting $R_1 \cap R_5 \neq \emptyset$. *Case 1.2:* Assume $cross(R_5, R_1)$. We still have $A(R_1) \cap A(R_4) = \emptyset$. But $A(R_5) \subseteq A(R_1)$ implies $A(R_4) \cap A(R_5) = \emptyset$ contradicting $R_4 \cap R_5 \neq \emptyset$. *Case 2:* Assume $cross(R_4, R_3)$. $cross(R_5, R_4)$ implies $cross(R_5, R_3)$ contradicting $R_3 \cap R_5 = \emptyset$. Thus, assume $cross(R_4, R_5)$. If $cross(R_2, R_5)$ then $cross(R_3, R_5)$ contradicting $R_3 \cap R_5 = \emptyset$. If $cross(R_5, R_2)$ then $\emptyset \neq B(R_2) \subseteq B(R_3) \cap B(R_5)$. As $R_3 \cap R_5 = \emptyset$, it must hold $A(R_3) \cap A(R_5) = \emptyset$. But $A(R_4) \subseteq A(R_5)$. We finally get $A(R_3) \cap A(R_4) = \emptyset$ contradicting $R_3 \cap R_4 \neq \emptyset$. We conclude that gem cannot be a cs-graph.

4. We assume that star is a cs-graph, i.e. star $=_{iso} \mathcal{G}(M)$ for a matrix M. We have $V(M) = \{R_{i,A}, R_{i,B} \mid i \in [3]\}$ and $E(M) = \{\{R_{i,A}, R_{i,B}\} \mid i \in [3]\} \cup \{\{R_{1,A}, R_{2,A}\}, \{R_{2,A}, R_{3,A}\}, \{R_{1,A}, R_{3,A}\}\}$. As $K_4 \not\leq_{iso}$ star, only cross situations are possible. W.l.o.g. we assume $cross(R_{1,A}, R_{2,A})$. *Case 1:* Assume $cross(R_{2,A}, R_{3,A})$. $cross(R_{2,A}, R_{2,B})$ implies $cross(R_{1,A}, R_{2,B})$ which contradicts $R_{1,A} \cap R_{2,B} = \emptyset$. $cross(R_{2,B}, R_{2,A})$ implies $cross(R_{2,B}, R_{3,A})$ contradicting $R_{2,B} \cap R_{3,A} = \emptyset$. *Case 2:* Assume $cross(R_{3,A}, R_{2,A})$. *Case 2.1:* Assume $cross(R_{1,A}, R_{3,A})$. $cross(R_{3,A}, R_{3,B})$ implies $cross(R_{1,A}, R_{3,B})$ contradicting $R_{1,A} \cap R_{3,B} = \emptyset$. $cross(R_{3,B}, R_{3,A})$ implies $cross(R_{3,B}, R_{2,A})$ contradicting $R_{2,A} \cap R_{3,B} = \emptyset$. *Case 2.2:* Assume $cross(R_{3,A}, R_{1,A})$. $cross(R_{1,A}, R_{1,B})$ implies $cross(R_{3,A}, R_{1,B})$ contradicting $R_{3,A} \cap R_{1,B} = \emptyset$. $cross(R_{1,B}, R_{1,A})$ implies $cross(R_{1,B}, R_{2,A})$ contradicting $R_{2,A} \cap R_{1,B} = \emptyset$. We conclude that star cannot be a cs-graph.

5. We assume that watch is a cs-graph, i.e. watch $=_{iso} \mathcal{G}(M)$ for a matrix M. We have $V(M) = \{R_1, \ldots, R_6\}$ and $E(M) = \{\{R_5, R_2\}, \{R_2, R_1\}, \{R_2, R_3\}, \{R_2, R_4\}, \{R_1, R_3\}, \{R_4, R_3\}, \{R_3, R_6\}\}$. As $K_4 \not\leq_{iso}$ watch, only cross situations are possible. W.l.o.g. we assume $cross(R_1, R_2)$. *Case 1:* Assume $cross(R_2, R_3)$. $cross(R_2, R_5)$ implies $cross(R_1, R_5)$ contradicting $R_1 \cap R_5 = \emptyset$. $cross(R_5, R_2)$ implies $cross(R_5, R_3)$ contradicting $R_3 \cap R_5 = \emptyset$. *Case 2:* Assume $cross(R_3, R_2)$. *Case 2.1:* Assume $cross(R_1, R_3)$. $cross(R_3, R_6)$ implies $cross(R_1, R_6)$ contradicting $R_1 \cap R_6 = \emptyset$. $cross(R_6, R_3)$ implies $cross(R_6, R_2)$ contradicting $R_2 \cap R_6 = \emptyset$. *Case 2.2:* Assume $cross(R_3, R_1)$. $cross(R_2, R_4)$ implies $cross(R_1, R_4)$ contradicting $R_1 \cap R_4 = \emptyset$. Thus, assume $cross(R_4, R_2)$. If $cross(R_3, R_4)$ then $R_2 \cap R_3 \subseteq R_4$ implying $R_1 \cap R_4 \neq \emptyset$, a contradiction. $cross(R_4, R_3)$ implies $cross(R_4, R_1)$, but again, then we have $R_1 \cap R_4 \neq \emptyset$, a contradiction. We conclude that watch cannot be a cs-graph. $\square$

3 Beautiful graphs

We have seen in the last section, that **csg** does not contain all graphs, i.e. covers have structure. As squares and odd holes are "forbidden", the previous results motivate the following definition:

Definition 3.1. A graph is *beautiful* iff every induced subgraph is a cs-graph.

Clearly, from Theorem 2.3 we obtain:

Theorem 3.2. *Every beautiful graph is a square-free Berge graph.* $\square$

The opposite is not true, as e.g. star is square-free and Berge, but not beautiful. A comparison with known classes of perfect graphs (see e.g. [6, 9] and Figure 3 below comparing the cs-graphs, K_4-free cs-graphs and the class **beautiful** of beautiful graphs with well-known classes of square-free perfect graphs, namely interval, split, threshold, triangulated and trivially perfect graphs) yields, that the class of beautiful graphs **beautiful** is a new class of perfect graphs. In Figure 3 we list the interesting class of K_4-free cs-graphs, because such graphs cannot be represented by matrices containing spade situations. We conjecture that they coincide with the class of K_4-free beautiful graphs. We state without proof (for space reasons), that the list of forbidden induced subgraphs of beautiful graphs in Theorem 2.3 is complete up to connected graphs of order $n \leq 7$.

	interval	**split**	**threshold**	**triangulated**	**triv.perfect**
beautiful	$\not\supseteq$, gem $\not\subseteq$, C_6	$\not\supseteq$, star $\not\subseteq$, $\overline{C}_4$	$\not\supseteq$, gem $\not\subseteq$, $\overline{C}_4$	$\not\supseteq$, gem $\not\subseteq$, C_6	$\not\supseteq$, star $\not\subseteq$, P_4
K_4-**free csg**	$\not\supseteq$, gem $\not\subseteq$, C_6	$\not\supseteq$, star $\not\subseteq$, $\overline{C}_4$	$\not\supseteq$, gem $\not\subseteq$, $\overline{C}_4$	$\not\supseteq$, gem $\not\subseteq$, C_6	$\not\supseteq$, star $\not\subseteq$, P_4
csg	$\not\supseteq$, gem $\not\subseteq$, C_6	$\not\supseteq$, star $\not\subseteq$, $\overline{C}_4$	$\not\supseteq$, gem $\not\subseteq$, $\overline{C}_4$	$\not\supseteq$, gem $\not\subseteq$, C_6	$\not\supseteq$, star $\not\subseteq$, P_4

Figure 3: Comparisons of graph classes

We explore the structure of beautiful graphs and make progress towards a characterization in the spirit of Conforti, Cornuéjols and Vušković [4]. Recall their characterization/decomposition theorem of square-free perfect graphs:

Fact 3.3. *A square-free perfect graph is bipartite or the line graph of a bipartite graph or has a star cutset or a 2-join.*

We are able to give characterizations of the beautiful square-free bipartite graphs (3.1) and the beautiful line graphs of square-free bipartite graphs (3.2).

3.1 Characterization of beautiful sqr.-free bipartite graphs

Proposition 3.4. *Every square-free bipartite graph is a cs-graph.*

Proof. Let $G := (U \dot{\cup} V, E)$ be square-free and bipartite. W.l.o.g. assume $U = [m]$ and $V = [n]$. Define the $m \times n$-matrix I by $I_{u,v} := 1$, if $\{u, v\} \in E$, and $I_{u,v} := 0$ otherwise, $u \in U$, $v \in V$. Let

$$M \quad := \quad \begin{pmatrix} 0 & E_n \\ E_m & I \end{pmatrix}$$

Consider any $R = A \times B \in V(M)$. If R covers elements in E_m, then necessarily $|A| = 1$. There exists $u \in [m]$, and $B = \{m + v \mid v \in [n], \{u, v\} \in E\}$. If R covers elements in E_n, then necessarily $|B| = 1$. There exists $v \in [n]$, and $A = \{n + u \mid u \in [m], \{u, v\} \in E\}$. Suppose, R covers only elements in I. Then necessarily, $|A|, |B| \geq 2$. Then there exist distinct $u_1, u_2 \in A$, and distinct $v_1, v_2 \in B$ such that $I_{u_i,v_j} = 1$, $i \in [2]$, $j \in [2]$. This means $\{u_1, v_1\}, \{v_1, u_2\}, \{u_2, v_2\}, \{v_2, u_1\} \in E$. As G is bipartite, we have $\{u_1, u_2\}, \{v_1, v_2\} \notin E$. Thus, $C_4 \leq_{iso} G$, a contradiction. We conclude $G =_{iso} \mathcal{G}(M)$. $\square$

As every induced subgraph of a square-free bipartite graph is square-free bipartite, from Proposition 3.4 we immediately obtain:

Theorem 3.5. *Every square-free bipartite graph is beautiful.* □

3.2 Characterization of beautiful line graphs of square-free bipartite graphs

Now we completely describe square-free line graphs of bipartite graphs, i.e. we consider line graphs of square-free bipartite graphs. Here, the situation is more complicated.
We begin by fixing some notation. In this section, we let $\tilde{G} := (U_l \cup U_r, \tilde{E})$ be a square-free bipartite graph, and we let $G := L(\tilde{G}) = (V, E)$, $V := \tilde{E}$, be its line graph. For $u \in U_l$ define $K_u^l := \{e \in V \mid u \in e\}$. K_u^r is defined analogously for $u \in U_r$. Each K_u^l is a clique in G and $\{K_u^l \mid u \in U_l\}$ is a partition of V, the *left clique partition of* G. The *right clique partition* is defined analogously. We prove all results for the *left side* only, but of course, they also hold for the *right side*. We need the following claim:

Claim 3.6. *Let $u, u' \in U_l$, $u \neq u'$, be arbitrary. Then between K_u^l and $K_{u'}^l$ there is at most one edge.*

Proof. We assume the opposite. Let $e_1, e_2 \in K_u^l$, $e_1 \neq e_2$, and let $d \in K_{u'}^l$, such that $\{e_1, d\}, \{e_2, d\} \in E$. Then there exist distinct $v_1, v_2 \in U_r$ such that $e_i = \{u, v_i\}$, $i \in [2]$. As $\{e_1, d\} \in E$, we obtain $d = \{u, v_1\}$, and also $d = \{u, v_2\}$ by $\{e_2, d\} \in E$, a contradiction.
Now we assume $e_1, e_2 \in K_u^l$, $e_1 \neq e_2$, and $d_1, d_2 \in K_{u'}^l$, $d_1 \neq d_2$, such that $\{e_1, d_1\}, \{e_2, d_2\} \in E$. By the argument above, we have $\{e_1, d_2\}, \{e_2, d_1\} \notin E$. As $\{e_1, e_2\}, \{d_1, d_2\} \in E$ we get $C_4 \leq_{iso} G$, again a contradiction. We conclude that there is at most one edge between K_u^l and $K_{u'}^l$. □

Definition 3.7. For $u \in U_l$ define the set of *connection nodes* as

$$B_u^l \quad := \quad \{e \in K_u^l \mid \exists u' \in U_l \colon u \neq u', e \text{ adjacent to } K_{u'}^l\}$$

Lemma 3.8. *Assume that G is beautiful. Then the following statements hold:*

1. *Assume there exist distinct $u, u' \in U_l$, distinct $e_1, e_2 \in K_u^l$, and $d \in K_{u'}^l$ such that $\{d, e_1\} \in E$. Let $G =_{iso} \mathcal{G}(M)$ for a matrix M. If $R(v)$ denotes the corresponding nonextendible combinatorial rectangle of $v \in V$ in M, then we must have $cross\{R(e_1), R(e_2)\}$ and $cross\{R(e_1), R(d)\}$.*

2. *In each clique K_u^l there exist at most two nodes adjacent to other cliques $K_\cdot^l$. Especially, we must have $|B_u^l| \leq 2$ for each $u \in U_l$.*

3. *Let $u_i \in U_l$ be pairwise distinct, and let $e_i \in K_{u_i}^l$, $i \in [3]$. Then the set of nodes $\{e_i \mid i \in [3]\}$ cannot form a triangle in G.*

4. *$\mathcal{G}(\bigcup_{u \in U_l} B_u^l)$ is bipartite.*

Proof. 1. We assume $spade\{R(e_1), R(e_2)\}$. By Lemma 2.2 (3) there exist distinct $g_1, g_2 \in V$ such that $\{e_1, e_2, g_1, g_2\}$ is a K_4 in G. By Claim 3.6 we get $g_1, g_2 \in K_u^l$. In case $cross\{R(e_1), R(d)\}$ we must have $\{d, e_1\}, \{d, g_1\} \in E$ or

$\{d, e_1\}, \{d, g_2\} \in E$, which is impossible by Claim 3.6. In case $spade\{R(e_1), R(d)\}$ by Lemma 2.2 (3) there exist distinct $h_1, h_2 \in V$ such that $\{e_1, d, h_1, h_2\}$ is a K_4 in G. In addition, the nodes g_1, g_2, h_1, h_2 are pairwise distinct. W.l.o.g. $\{h_1, e_1\}, \{h_1, g_1\} \in E$. The case $\{h_1, e_1\}, \{h_1, g_2\} \in E$ is analogous. If $h_1 \in K_u^l$, then $\{d, e_1\}, \{d, h_1\} \in E$ contradicting Claim 3.6. If $h_1 \notin K_u^l$, then there exists $u'' \in U_l$, $u \neq u''$, such that $h_1 \in K_{u''}^l$. But then $\{h_1, e_1\}, \{h_1, g_1\} \in E$ again contradicts Claim 3.6. We conclude that the situation $spade\{R(e_1), R(e_2)\}$ cannot occur. By Lemma 2.2 (1) we obtain $cross\{R(e_1), R(e_2)\}$ proving the first statement.
Now, we assume $spade\{R(e_1), R(d)\}$. By Lemma 2.2 (3) there exist distinct $g_1, g_2 \in V$ such that $\{e_1, d, g_1, g_2\}$ is a K_4 in G. By Claim 3.6 there must exist $u_1, u_2 \in U_l$, u, u', u_1, u_2 pairwise distinct, such that $g_1 \in K_{u_1}^l$ and $g_2 \in K_{u_2}^l$. We saw in the first part of this proof, that we must have a $cross\{R(e_1), R(e_2)\}$ situation between e_1 and e_2. This implies the situation $cross\{R(e_2), R(g_1)\}$ or $cross\{R(e_2), R(g_2)\}$. But both $\{e_2, g_1\} \in E$ or $\{e_2, g_2\} \in E$ together with $\{e_1, g_1\}, \{e_1, g_2\} \in E$ contradict Claim 3.6. We conclude $cross\{R(e_1), R(d)\}$.

2. We assume the opposite. Let $u, u_1, u_2, u_3 \in U_l$ be pairwise distinct, and let $e_1, e_2, e_3 \in K_u^l$ be pairwise distinct. Let $g_i \in K_{u_i}^l$, such that $\{g_i, e_i\} \in E$, $i \in [3]$. By (1) we only have cross situations $cross\{R(g_i), R(e_i)\}$, $i \in [3]$, $cross\{R(e_1), R(e_2)\}$, $cross\{R(e_1), R(e_3)\}$ and also $cross\{R(e_2), R(e_3)\}$. W.l.o.g. we can assume $cross(R(g_1), R(e_1))$. Then $cross(R(e_2), R(e_1))$ as otherwise the case $R(g_1) \cap R(e_2) \neq \emptyset$ would imply $\{g_1, e_2\} \in E$ contradicting Claim 3.6. By an analogous argument we get $cross(R(e_2), R(g_2))$ and $cross(R(e_2), R(e_3))$. $B(R(e_1)) \cap B(\ R(e_3)) = \emptyset$ cannot be the case, as $\{e_1, e_3\} \in E$ (K_u^l is a clique). But $B(R(e_1)) \cap B(R(e_3)) \neq \emptyset$ implies $R(e_3) \cap R(g_1) \neq \emptyset$ and thus, $\{e_3, g_1\} \in E$, again contradicting Claim 3.6. We conclude that in each clique K_u^l there are at most two nodes adjacent to other cliques $K_{\cdot}^l$.

3. We assume the opposite. Then there exist $v, v_1, v_2, v_3 \in U_r$ pairwise distinct, such that $\{u_i, v_i\}, \{u_i, v\} \in \tilde{E}$, $i \in [3]$. Thus, in K_v^r there exist more than two nodes adjacent to other cliques in contradiction to (2), which also holds for the right clique partition.

4. We assume, that the induced subgraph $D := G(\bigcup_{u \in U_l} B_u^l)$ is not bipartite. Then D contains an odd cycle. As G is beautiful, also D is beautiful. One can show by induction on the cycle length, that a Berge graph containing an odd cycle as a subgraph (not necessarily induced) contains a triangle. Thus, D contains a triangle $\{e_1, e_2, e_3\}$. Each node e_i must lie in a separate clique by Claim 3.6. But this contradicts (3). We conclude that D must be bipartite. $\square$

The derivations above (Lemma 3.8) motivate the following definition:

Definition 3.9. G has the *property $\mathcal{B}$* if

- $|B_u^l| \leq 2$ for each $u \in U_l$, and
- $G(\bigcup_{u \in U_l} B_u^l)$ is bipartite.

Lemma 3.10. *If G has property $\mathcal{B}$, then G is a cs-graph.*

Proof. Let $G = L(\tilde{G})$ be the line graph of a square-free bipartite graph $\tilde{G} = (U_l \cup U_r, \tilde{E})$, where G has property $\mathcal{B}$. W.l.o.g. assume $U_l = [m]$. Let $\{K_u^l \mid u \in U_l\}$ be the left clique partition of G. Define $s_u := |K_u^l|$ and $s := \max\{s_u \mid u \in U_l\}$. We have to define a matrix M such that $G =_{iso} \mathcal{G}(M)$. Define the block matrix $M := (M_{i,j})_{i,j \in [m]}$, where each block $M_{i,j}$ is an $s \times s$-matrix over $\{0, 1\}$. In the diagonal, the left cliques are represented, i.e. $M_{i,i} := {}_{\mathrm{rep}}\mathrm{K}_{s_i}^{(s)}$, $i \in [m]$. See Subsection 1.4 for the definition of ${}_{\mathrm{rep}}\mathrm{K}_{s_i}^{(s)}$. By Lemma 3.8 (4) we know that $\mathcal{G}(\bigcup_{u \in U_l} B_u^l)$ is bipartite. Thus, there exists a 2-coloring $c\colon \bigcup_{u \in U_l} B_u^l \to [2]$. For each $B_u^l \neq \emptyset$, we can now define its elements. If $e \in B_u^l$ and $c(e) = b$, define $e_u^b := e$, $b \in [2]$. Trivially, $c(e_u^b) = b$. Now we can define the nondiagonal blocks in M: For distinct $i, j \in [m]$, if e_i^1 is adjacent to e_j^2, then define

$$M_{i,j} \quad := \quad E_{1,1} := \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$$

Otherwise, let $M_{i,j} := (0)$. Note, that there does not exist any nonextendible rectangle R in M, such that R covers elements in blocks $M_{i,j}, M_{r,s}$ with $i \neq r$, $j \neq s$. Assume the contrary. We distinguish three cases: In case $i = j$ and $r = s$ the left cliques K_i^l and K_r^l would be connected by two edges $\{e_i^1, e_r^2\}, \{e_i^2, e_r^1\} \in E$, which is impossible. In case $i = j$ and $r \neq s$ by construction of M, there would exist edges $\{e_i^1, e_r^2\}, \{e_s^1, e_r^2\}, \{e_s^1, e_i^2\} \in E$ implying $C_4 \leq_{iso} G$, because $\{e_i^1, e_i^2\} \in E$ and $\{e_i^1, e_s^1\}, \{e_i^2, e_r^2\} \notin E$, as c is a 2-coloring. But this is impossible, as G is square-free. Similarly, in case $i \neq j$ and $r \neq s$, there would exist edges $\{e_i^1, e_j^2\}, \{e_j^2, e_r^1\}, \{e_r^1, e_s^2\}, \{e_s^2, e_i^1\} \in E$, again forming an induced C_4 in G. Thus, such a rectangle R cannot exist, and we conclude $G =_{iso} \mathcal{G}(M)$. □

Every induced subgraph of the line graph of a square-free bipartite graph, which has property $\mathcal{B}$, is also the line graph of a square-free bipartite graph, which has property $\mathcal{B}$. Thus, by Lemma 3.10 we have:

Theorem 3.11. *If G has property $\mathcal{B}$, then G is beautiful.* □

As a technical intermediate characterization by Lemma 3.8 (2) and (4), and Theorem 3.11 we get:

Theorem 3.12. *G is beautiful iff G has property $\mathcal{B}$.* □

But what do these graphs look like? In U_r we can safely ignore isolated nodes. We delete nodes of degree one obtaining U_r'. We also delete nodes in U_l which have become isolated. Call the new set U_l'. The property of $G(\bigcup_{u \in U_l} B_u^l)$ being bipartite implies that in U_r' nodes of degree ≥ 3 do not exist (otherwise, one would have a triangle). If we restrict $\tilde{G}$ on U_l' and U_r', all nodes on the right side have degree two while all nodes on the left have degree one. This graph consists of disjoint cycles of even length ≥ 6 and paths of even length ≥ 2. Thus, the corresponding line graph consists of cycles of even length and paths of odd length. We color the edges of G red and green such that end edges are colored green. The leaves in $\tilde{G}$ induce additional cliques in the line graph. These are cliques of arbitrary size which are attached to the start or end nodes of a path or contain a single red edge of a path or a cycle and only additional nodes and edges. We call such graphs *Odd Paths and Even Cycles of Cliques* graph, see e.g. Figure 4.

Thus, our main theorem reads as follows:

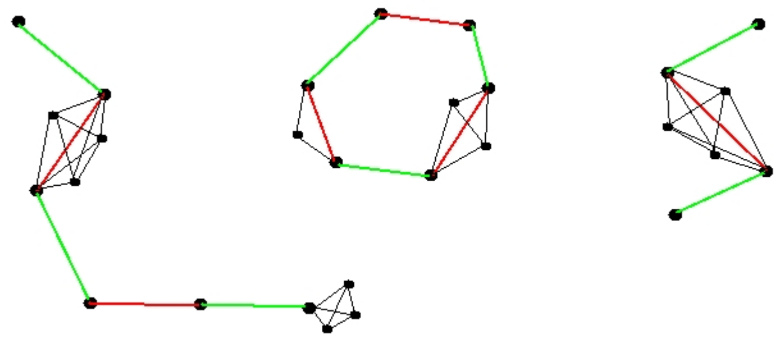

Figure 4: Odd Paths and Even Cycles of Cliques

Theorem 3.13. *A line graph of a square-free bipartite graph is beautiful iff it is an* Odd Paths and Even Cycles of Cliques *graph.* □

Acknowledgements I would like to thank Martin Dietzfelbinger for many fruitful discussions and for his contribution to Theorem 3.13.
I would also like to thank Andreas Brandstädt and Michael Stiebitz for their suggestions concerning the presentation of this paper, Jacobo Torán for interesting discussions, and Thanh Minh Hoang and Fabian Wagner for careful proofreading.

References

[1] N. Alon, E. Lubetzky, The shannon capacity of a graph and the independence numbers of its powers, IEEE Transactions on Information Theory 52 (5) (2006) 2172–2176.

[2] C. Berge, Färbung von Graphen, deren sämtliche bzw. deren ungerade Kreise starr sind (Zusammenfassung), Wissenschaftliche Zeitschrift, Martin Luther Universität Halle-Wittenberg, Mathematisch-Naturwissenschaftliche Reihe (1961) 114–115.

[3] M. Chudnovsky, N. Robertson, P. Seymour, R. Thomas, The strong perfect graph theorem, Annals of Mathematics 164 (2006) 51–229.

[4] M. Conforti, G. Cornuéjols, K. Vušković, Square-free perfect graphs, J. Comb. Theory, Ser. B 90 (2) (2004) 257–307.

[5] R. Diestel, Graph Theory, vol. 173 of Graduate Texts in Mathematics, 3rd ed., Springer-Verlag, Heidelberg, 2005.

[6] S. Hougardy, Classes of perfect graphs, Discrete Mathematics 306 (19-20) (2006) 2529–2571.

[7] E. Kushilevitz, N. Nisan, Communication Complexity, Cambridge University Press, 1997.

[8] L. Lovász, On the shannon capacity of a graph, IEEE Trans. Inform. Theory IT-25 (1979) 1–7.

[9] T. A. McKee, F. R. McMorris, Topics in Intersection Graph Theory, SIAM Monographs on Discrete Mathematics and Applications, 1999.

[10] J. L. Ramírez-Alfonsín, B. A. Reed, Perfect Graphs, Wiley, 2001.

[11] C. E. Shannon, A mathematical theory of communication, Bell Sys. Tech. Journal 27 (1948) 379–423 and 623–656.

[12] C. E. Shannon, The zero-error capacity of a noisy channel, IRE Trans. Inform. Theory IT-2 (1965) 8–19.

[13] A. C.-C. Yao, Some complexity questions related to distributive computing (preliminary report), in: Conference Record of the Eleventh Annual ACM Symposium on Theory of Computing, 30 April-2 May, 1979, Atlanta, Georgia, USA, 1979.

Liste der bisher erschienenen Ulmer Informatik-Berichte
Einige davon sind per FTP von `ftp.informatik.uni-ulm.de` erhältlich
Die mit * markierten Berichte sind vergriffen

List of technical reports published by the University of Ulm
Some of them are available by FTP from `ftp.informatik.uni-ulm.de`
Reports marked with * are out of print

91-01 *Ker-I Ko, P. Orponen, U. Schöning, O. Watanabe*
Instance Complexity

91-02* *K. Gladitz, H. Fassbender, H. Vogler*
Compiler-Based Implementation of Syntax-Directed Functional Programming

91-03* *Alfons Geser*
Relative Termination

91-04* *J. Köbler, U. Schöning, J. Toran*
Graph Isomorphism is low for PP

91-05 *Johannes Köbler, Thomas Thierauf*
Complexity Restricted Advice Functions

91-06* *Uwe Schöning*
Recent Highlights in Structural Complexity Theory

91-07* *F. Green, J. Köbler, J. Toran*
The Power of Middle Bit

91-08* *V.Arvind, Y. Han, L. Hamachandra, J. Köbler, A. Lozano, M. Mundhenk, A. Ogiwara, U. Schöning, R. Silvestri, T. Thierauf*
Reductions for Sets of Low Information Content

92-01* *Vikraman Arvind, Johannes Köbler, Martin Mundhenk*
On Bounded Truth-Table and Conjunctive Reductions to Sparse and Tally Sets

92-02* *Thomas Noll, Heiko Vogler*
Top-down Parsing with Simulataneous Evaluation of Noncircular Attribute Grammars

92-03 *Fakultät für Informatik*
17. Workshop über Komplexitätstheorie, effiziente Algorithmen und Datenstrukturen

92-04* *V. Arvind, J. Köbler, M. Mundhenk*
Lowness and the Complexity of Sparse and Tally Descriptions

92-05* *Johannes Köbler*
Locating P/poly Optimally in the Extended Low Hierarchy

92-06* *Armin Kühnemann, Heiko Vogler*
Synthesized and inherited functions -a new computational model for syntax-directed semantics

92-07* *Heinz Fassbender, Heiko Vogler*
A Universal Unification Algorithm Based on Unification-Driven Leftmost Outermost Narrowing

92-08* *Uwe Schöning*
On Random Reductions from Sparse Sets to Tally Sets

92-09* *Hermann von Hasseln, Laura Martignon*
Consistency in Stochastic Network

92-10 *Michael Schmitt*
A Slightly Improved Upper Bound on the Size of Weights Sufficient to Represent Any Linearly Separable Boolean Function

92-11 *Johannes Köbler, Seinosuke Toda*
On the Power of Generalized MOD-Classes

92-12 *V. Arvind, J. Köbler, M. Mundhenk*
Reliable Reductions, High Sets and Low Sets

92-13 *Alfons Geser*
On a monotonic semantic path ordering

92-14* *Joost Engelfriet, Heiko Vogler*
The Translation Power of Top-Down Tree-To-Graph Transducers

93-01 *Alfred Lupper, Konrad Froitzheim*
AppleTalk Link Access Protocol basierend auf dem Abstract Personal Communications Manager

93-02 *M.H. Scholl, C. Laasch, C. Rich, H.-J. Schek, M. Tresch*
The COCOON Object Model

93-03 *Thomas Thierauf, Seinosuke Toda, Osamu Watanabe*
On Sets Bounded Truth-Table Reducible to P-selective Sets

93-04 *Jin-Yi Cai, Frederic Green, Thomas Thierauf*
On the Correlation of Symmetric Functions

93-05 *K.Kuhn, M.Reichert, M. Nathe, T. Beuter, C. Heinlein, P. Dadam*
A Conceptual Approach to an Open Hospital Information System

93-06 *Klaus Gaßner*
Rechnerunterstützung für die konzeptuelle Modellierung

93-07 *Ullrich Keßler, Peter Dadam*
Towards Customizable, Flexible Storage Structures for Complex Objects

94-01 *Michael Schmitt*
On the Complexity of Consistency Problems for Neurons with Binary Weights

94-02 *Armin Kühnemann, Heiko Vogler*
A Pumping Lemma for Output Languages of Attributed Tree Transducers

94-03 *Harry Buhrman, Jim Kadin, Thomas Thierauf*
On Functions Computable with Nonadaptive Queries to NP

94-04 *Heinz Faßbender, Heiko Vogler, Andrea Wedel*
Implementation of a Deterministic Partial E-Unification Algorithm for Macro Tree Transducers

94-05 *V. Arvind, J. Köbler, R. Schuler*
On Helping and Interactive Proof Systems

94-06 *Christian Kalus, Peter Dadam*
Incorporating record subtyping into a relational data model

94-07 *Markus Tresch, Marc H. Scholl*
A Classification of Multi-Database Languages

94-08 *Friedrich von Henke, Harald Rueß*
Arbeitstreffen Typtheorie: Zusammenfassung der Beiträge

94-09 *F.W. von Henke, A. Dold, H. Rueß, D. Schwier, M. Strecker*
Construction and Deduction Methods for the Formal Development of Software

94-10 *Axel Dold*
Formalisierung schematischer Algorithmen

94-11 *Johannes Köbler, Osamu Watanabe*
New Collapse Consequences of NP Having Small Circuits

94-12 *Rainer Schuler*
On Average Polynomial Time

94-13 *Rainer Schuler, Osamu Watanabe*
Towards Average-Case Complexity Analysis of NP Optimization Problems

94-14 *Wolfram Schulte, Ton Vullinghs*
Linking Reactive Software to the X-Window System

94-15 *Alfred Lupper*
Namensverwaltung und Adressierung in Distributed Shared Memory-Systemen

94-16 *Robert Regn*
Verteilte Unix-Betriebssysteme

94-17 *Helmuth Partsch*
Again on Recognition and Parsing of Context-Free Grammars:
Two Exercises in Transformational Programming

94-18 *Helmuth Partsch*
Transformational Development of Data-Parallel Algorithms: an Example

95-01 *Oleg Verbitsky*
On the Largest Common Subgraph Problem

95-02 *Uwe Schöning*
Complexity of Presburger Arithmetic with Fixed Quantifier Dimension

95-03 *Harry Buhrman,Thomas Thierauf*
The Complexity of Generating and Checking Proofs of Membership

95-04 *Rainer Schuler, Tomoyuki Yamakami*
Structural Average Case Complexity

95-05 *Klaus Achatz, Wolfram Schulte*
Architecture Indepentent Massive Parallelization of Divide-And-Conquer Algorithms

95-06 *Christoph Karg, Rainer Schuler*
Structure in Average Case Complexity

95-07 *P. Dadam, K. Kuhn, M. Reichert, T. Beuter, M. Nathe*
ADEPT: Ein integrierender Ansatz zur Entwicklung flexibler, zuverlässiger kooperierender Assistenzsysteme in klinischen Anwendungsumgebungen

95-08 *Jürgen Kehrer, Peter Schulthess*
Aufbereitung von gescannten Röntgenbildern zur filmlosen Diagnostik

95-09 *Hans-Jörg Burtschick, Wolfgang Lindner*
On Sets Turing Reducible to P-Selective Sets

95-10 *Boris Hartmann*
Berücksichtigung lokaler Randbedingung bei globaler Zieloptimierung mit neuronalen Netzen am Beispiel Truck Backer-Upper

95-12 *Klaus Achatz, Wolfram Schulte*
Massive Parallelization of Divide-and-Conquer Algorithms over Powerlists

95-13 *Andrea Mößle, Heiko Vogler*
Efficient Call-by-value Evaluation Strategy of Primitive Recursive Program Schemes

95-14 *Axel Dold, Friedrich W. von Henke, Holger Pfeifer, Harald Rueß*
A Generic Specification for Verifying Peephole Optimizations

96-01 *Ercüment Canver, Jan-Tecker Gayen, Adam Moik*
Formale Entwicklung der Steuerungssoftware für eine elektrisch ortsbediente Weiche mit VSE

96-02 *Bernhard Nebel*
Solving Hard Qualitative Temporal Reasoning Problems: Evaluating the Efficiency of Using the ORD-Horn Class

96-03 *Ton Vullinghs, Wolfram Schulte, Thilo Schwinn*
An Introduction to TkGofer

96-04 *Thomas Beuter, Peter Dadam*
Anwendungsspezifische Anforderungen an Workflow-Mangement-Systeme am Beispiel der Domäne Concurrent-Engineering

96-05 *Gerhard Schellhorn, Wolfgang Ahrendt*
Verification of a Prolog Compiler - First Steps with KIV

96-06 *Manindra Agrawal, Thomas Thierauf*
Satisfiability Problems

96-07 *Vikraman Arvind, Jacobo Torán*
A nonadaptive NC Checker for Permutation Group Intersection

96-08 *David Cyrluk, Oliver Möller, Harald Rueß*
An Efficient Decision Procedure for a Theory of Fix-Sized Bitvectors with Composition and Extraction

96-09 *Bernd Biechele, Dietmar Ernst, Frank Houdek, Joachim Schmid, Wolfram Schulte*
Erfahrungen bei der Modellierung eingebetteter Systeme mit verschiedenen SA/RT–Ansätzen

96-10 *Falk Bartels, Axel Dold, Friedrich W. von Henke, Holger Pfeifer, Harald Rueß*
Formalizing Fixed-Point Theory in PVS

96-11 *Axel Dold, Friedrich W. von Henke, Holger Pfeifer, Harald Rueß*
Mechanized Semantics of Simple Imperative Programming Constructs

96-12 *Axel Dold, Friedrich W. von Henke, Holger Pfeifer, Harald Rueß*
Generic Compilation Schemes for Simple Programming Constructs

96-13 *Klaus Achatz, Helmuth Partsch*
From Descriptive Specifications to Operational ones: A Powerful Transformation Rule, its Applications and Variants

97-01 *Jochen Messner*
Pattern Matching in Trace Monoids

97-02 *Wolfgang Lindner, Rainer Schuler*
A Small Span Theorem within P

97-03 *Thomas Bauer, Peter Dadam*
A Distributed Execution Environment for Large-Scale Workflow Management Systems with Subnets and Server Migration

97-04 *Christian Heinlein, Peter Dadam*
Interaction Expressions - A Powerful Formalism for Describing Inter-Workflow Dependencies

97-05 *Vikraman Arvind, Johannes Köbler*
On Pseudorandomness and Resource-Bounded Measure

97-06 *Gerhard Partsch*
Punkt-zu-Punkt- und Mehrpunkt-basierende LAN-Integrationsstrategien für den digitalen Mobilfunkstandard DECT

97-07 *Manfred Reichert, Peter Dadam*
$ADEPT_{flex}$ - Supporting Dynamic Changes of Workflows Without Loosing Control

97-08 *Hans Braxmeier, Dietmar Ernst, Andrea Mößle, Heiko Vogler*
The Project NoName - A functional programming language with its development environment

97-09 *Christian Heinlein*
Grundlagen von Interaktionsausdrücken

97-10 *Christian Heinlein*
Graphische Repräsentation von Interaktionsausdrücken

97-11 *Christian Heinlein*
Sprachtheoretische Semantik von Interaktionsausdrücken

97-12 *Gerhard Schellhorn, Wolfgang Reif*
Proving Properties of Finite Enumerations: A Problem Set for Automated Theorem Provers

97-13 *Dietmar Ernst, Frank Houdek, Wolfram Schulte, Thilo Schwinn*
Experimenteller Vergleich statischer und dynamischer Softwareprüfung für eingebettete Systeme

97-14 *Wolfgang Reif, Gerhard Schellhorn*
Theorem Proving in Large Theories

97-15 *Thomas Wennekers*
Asymptotik rekurrenter neuronaler Netze mit zufälligen Kopplungen

97-16 *Peter Dadam, Klaus Kuhn, Manfred Reichert*
Clinical Workflows - The Killer Application for Process-oriented Information Systems?

97-17 *Mohammad Ali Livani, Jörg Kaiser*
EDF Consensus on CAN Bus Access in Dynamic Real-Time Applications

97-18 *Johannes Köbler,Rainer Schuler*
Using Efficient Average-Case Algorithms to Collapse Worst-Case Complexity Classes

98-01 *Daniela Damm, Lutz Claes, Friedrich W. von Henke, Alexander Seitz, Adelinde Uhrmacher, Steffen Wolf*
Ein fallbasiertes System für die Interpretation von Literatur zur Knochenheilung

98-02 *Thomas Bauer, Peter Dadam*
Architekturen für skalierbare Workflow-Management-Systeme - Klassifikation und Analyse

98-03 *Marko Luther, Martin Strecker*
A guided tour through *Typelab*

98-04 *Heiko Neumann, Luiz Pessoa*
Visual Filling-in and Surface Property Reconstruction

98-05 *Ercüment Canver*
Formal Verification of a Coordinated Atomic Action Based Design

98-06 *Andreas Küchler*
On the Correspondence between Neural Folding Architectures and Tree Automata

98-07 *Heiko Neumann, Thorsten Hansen, Luiz Pessoa*
Interaction of ON and OFF Pathways for Visual Contrast Measurement

98-08 *Thomas Wennekers*
Synfire Graphs: From Spike Patterns to Automata of Spiking Neurons

98-09 *Thomas Bauer, Peter Dadam*
Variable Migration von Workflows in *ADEPT*

98-10 *Heiko Neumann, Wolfgang Sepp*
Recurrent V1 – V2 Interaction in Early Visual Boundary Processing

98-11 *Frank Houdek, Dietmar Ernst, Thilo Schwinn*
Prüfen von C–Code und Statmate/Matlab–Spezifikationen: Ein Experiment

98-12 *Gerhard Schellhorn*
Proving Properties of Directed Graphs: A Problem Set for Automated Theorem Provers

98-13 *Gerhard Schellhorn, Wolfgang Reif*
Theorems from Compiler Verification: A Problem Set for Automated Theorem Provers

98-14 *Mohammad Ali Livani*
SHARE: A Transparent Mechanism for Reliable Broadcast Delivery in CAN

98-15 *Mohammad Ali Livani, Jörg Kaiser*
Predictable Atomic Multicast in the Controller Area Network (CAN)

99-01 *Susanne Boll, Wolfgang Klas, Utz Westermann*
A Comparison of Multimedia Document Models Concerning Advanced Requirements

99-02 *Thomas Bauer, Peter Dadam*
Verteilungsmodelle für Workflow-Management-Systeme - Klassifikation und Simulation

99-03 *Uwe Schöning*
On the Complexity of Constraint Satisfaction

99-04 *Ercument Canver*
Model-Checking zur Analyse von Message Sequence Charts über Statecharts

99-05 *Johannes Köbler, Wolfgang Lindner, Rainer Schuler*
Derandomizing RP if Boolean Circuits are not Learnable

99-06 *Utz Westermann, Wolfgang Klas*
Architecture of a DataBlade Module for the Integrated Management of Multimedia Assets

99-07 *Peter Dadam, Manfred Reichert*
Enterprise-wide and Cross-enterprise Workflow Management: Concepts, Systems, Applications. Paderborn, Germany, October 6, 1999, GI–Workshop Proceedings, Informatik ’99

99-08 *Vikraman Arvind, Johannes Köbler*
Graph Isomorphism is Low for ZPP^{NP} and other Lowness results

99-09 *Thomas Bauer, Peter Dadam*
Efficient Distributed Workflow Management Based on Variable Server Assignments

2000-02 *Thomas Bauer, Peter Dadam*
Variable Serverzuordnungen und komplexe Bearbeiterzuordnungen im Workflow-Management-System ADEPT

2000-03 *Gregory Baratoff, Christian Toepfer, Heiko Neumann*
Combined space-variant maps for optical flow based navigation

2000-04 *Wolfgang Gehring*
Ein Rahmenwerk zur Einführung von Leistungspunktsystemen

2000-05 *Susanne Boll, Christian Heinlein, Wolfgang Klas, Jochen Wandel*
Intelligent Prefetching and Buffering for Interactive Streaming of MPEG Videos

2000-06 *Wolfgang Reif, Gerhard Schellhorn, Andreas Thums*
Fehlersuche in Formalen Spezifikationen

2000-07 *Gerhard Schellhorn, Wolfgang Reif (eds.)*
FM-Tools 2000: The 4th Workshop on Tools for System Design and Verification

2000-08 *Thomas Bauer, Manfred Reichert, Peter Dadam*
Effiziente Durchführung von Prozessmigrationen in verteilten Workflow-Management-Systemen

2000-09 *Thomas Bauer, Peter Dadam*
Vermeidung von Überlastsituationen durch Replikation von Workflow-Servern in ADEPT

2000-10 *Thomas Bauer, Manfred Reichert, Peter Dadam*
Adaptives und verteiltes Workflow-Management

2000-11 *Christian Heinlein*
Workflow and Process Synchronization with Interaction Expressions and Graphs

2001-01 *Hubert Hug, Rainer Schuler*
DNA-based parallel computation of simple arithmetic

2001-02 *Friedhelm Schwenker, Hans A. Kestler, Günther Palm*
3-D Visual Object Classification with Hierarchical Radial Basis Function Networks

2001-03 *Hans A. Kestler, Friedhelm Schwenker, Günther Palm*
RBF network classification of ECGs as a potential marker for sudden cardiac death

2001-04 *Christian Dietrich, Friedhelm Schwenker, Klaus Riede, Günther Palm*
Classification of Bioacoustic Time Series Utilizing Pulse Detection, Time and Frequency Features and Data Fusion

2002-01 *Stefanie Rinderle, Manfred Reichert, Peter Dadam*
Effiziente Verträglichkeitsprüfung und automatische Migration von Workflow-Instanzen bei der Evolution von Workflow-Schemata

2002-02 *Walter Guttmann*
Deriving an Applicative Heapsort Algorithm

2002-03 *Axel Dold, Friedrich W. von Henke, Vincent Vialard, Wolfgang Goerigk*
A Mechanically Verified Compiling Specification for a Realistic Compiler

2003-01 *Manfred Reichert, Stefanie Rinderle, Peter Dadam*
A Formal Framework for Workflow Type and Instance Changes Under Correctness Checks

2003-02 *Stefanie Rinderle, Manfred Reichert, Peter Dadam*
Supporting Workflow Schema Evolution By Efficient Compliance Checks

2003-03 *Christian Heinlein*
Safely Extending Procedure Types to Allow Nested Procedures as Values

2003-04 *Stefanie Rinderle, Manfred Reichert, Peter Dadam*
On Dealing With Semantically Conflicting Business Process Changes.

2003-05 *Christian Heinlein*
Dynamic Class Methods in Java

2003-06 *Christian Heinlein*
Vertical, Horizontal, and Behavioural Extensibility of Software Systems

2003-07 *Christian Heinlein*
Safely Extending Procedure Types to Allow Nested Procedures as Values
(Corrected Version)

2003-08 *Changling Liu, Jörg Kaiser*
Survey of Mobile Ad Hoc Network Routing Protocols)

2004-01 *Thom Frühwirth, Marc Meister (eds.)*
First Workshop on Constraint Handling Rules

2004-02 *Christian Heinlein*
Concept and Implementation of C+++, an Extension of C++ to Support User-Defined Operator Symbols and Control Structures

2004-03 *Susanne Biundo, Thom Frühwirth, Günther Palm(eds.)*
Poster Proceedings of the 27th Annual German Conference on Artificial Intelligence

2005-01 *Armin Wolf, Thom Frühwirth, Marc Meister (eds.)*
19th Workshop on (Constraint) Logic Programming

2005-02 *Wolfgang Lindner (Hg.), Universität Ulm , Christopher Wolf (Hg.) KU Leuven*
2. Krypto-Tag – Workshop über Kryptographie, Universität Ulm

2005-03 *Walter Guttmann, Markus Maucher*
Constrained Ordering

2006-01 *Stefan Sarstedt*
Model-Driven Development with ACTIVECHARTS, Tutorial

2006-02 *Alexander Raschke, Ramin Tavakoli Kolagari*
Ein experimenteller Vergleich zwischen einer plan-getriebenen und einer leichtgewichtigen Entwicklungsmethode zur Spezifikation von eingebetteten Systemen

2006-03 *Jens Kohlmeyer, Alexander Raschke, Ramin Tavakoli Kolagari*
Eine qualitative Untersuchung zur Produktlinien-Integration über Organisationsgrenzen hinweg

2006-04 *Thorsten Liebig*
Reasoning with OWL - System Support and Insights –

2008-01 *H.A. Kestler, J. Messner, A. Müller, R. Schuler*
On the complexity of intersecting multiple circles for graphical display

2008-02 *Manfred Reichert, Peter Dadam, Martin Jurisch,l Ulrich Kreher, Kevin Göser, Markus Lauer*
Architectural Design of Flexible Process Management Technology

2008-03 *Frank Raiser*
Semi-Automatic Generation of CHR Solvers from Global Constraint Automata

2008-04 *Ramin Tavakoli Kolagari, Alexander Raschke, Matthias Schneiderhan, Ian Alexander*
Entscheidungsdokumentation bei der Entwicklung innovativer Systeme für produktlinien-basierte Entwicklungsprozesse

2008-05 *Markus Kalb, Claudia Dittrich, Peter Dadam*
Support of Relationships Among Moving Objects on Networks

2008-06 *Matthias Frank, Frank Kargl, Burkhard Stiller (Hg.)*
WMAN 2008 – KuVS Fachgespräch über Mobile Ad-hoc Netzwerke

2008-07 *M. Maucher, U. Schöning, H.A. Kestler*
An empirical assessment of local and population based search methods with different degrees of pseudorandomness

2008-08 *Henning Wunderlich*
Covers have structure

Ulmer Informatik-Berichte
ISSN 0939-5091

Herausgeber:
Universität Ulm
Fakultät für Ingenieurwissenschaften und Informatik
89069 Ulm