Electronic Colloquium on Computational Complexity, Report No. 68 (2002)

Complexity of the Exact Domatic Number Problem and of the Exact Conveyor Flow Shop Problem

Tobias Riege* and Jörg Rothe†
Institut für Informatik
Heinrich-Heine-Universität Düsseldorf
40225 Düsseldorf, Germany

December 9, 2002

Abstract

We prove that the exact versions of the domatic number problem are complete for the levels of the boolean hierarchy over NP. The domatic number problem, which arises in the area of computer networks, is the problem of partitioning a given graph into a maximum number of disjoint dominating sets. This number is called the domatic number of the graph. We prove that the problem of determining whether or not the domatic number of a given graph is *exactly* one of k given values is complete for $\mathrm{BH}_{2k}(\mathrm{NP})$, the $2k$th level of the boolean hierarchy over NP. In particular, for $k = 1$, it is DP-complete to determine whether or not the domatic number of a given graph equals exactly a given integer. Note that $\mathrm{DP} = \mathrm{BH}_2(\mathrm{NP})$. We obtain similar results for the exact versions of the conveyor flow shop problem, which arises in real-world applications in the wholesale business, where warehouses are supplied with goods from a central storehouse. Our reductions apply Wagner's conditions sufficient to prove hardness for the levels of the boolean hierarchy over NP.

Key words: *Computational complexity; completeness; domatic number problem; conveyor flow shop problem; boolean hierarchy*

*Email: `riege@cs.uni-duesseldorf.de`.
†Email: `rothe@cs.uni-duesseldorf.de`.

ISSN 1433-8092

1 Introduction

A dominating set in an undirected graph G is a subset D of the vertex set $V(G)$ such that every vertex of $V(G)$ either belongs to D or is adjacent to some vertex in D. The domatic number problem is the problem of partitioning the vertex set $V(G)$ into a maximum number of disjoint dominating sets. This number, denoted by $\delta(G)$, is called the domatic number of G. The domatic number problem arises in various areas and scenarios. In particular, this problem is related to the task of distributing resources in a computer network, and also to the task of locating facilities in a communication network.

Suppose, for example, that resources are to be allocated in a computer network such that expensive services are quickly accessible in the immediate neighborhood of each vertex. If every vertex has only a limited capacity, then there is a bound on the number of resources that can be supported. In particular, if every vertex can serve a single resource only, then the maximum number of resources that can be supported equals the domatic number of the network graph. In the communication network scenario, n cities are linked via communication channels. A transmitting group is a subset of those cities that are able to transmit messages to every city in the network. Such a transmitting group is nothing else than a dominating set in the network graph, and the domatic number of this graph is the maximum number of disjoint transmitting groups in the network.

Motivated by these scenarios, the domatic number problem has been thoroughly investigated. Its decision version, denoted by DNP, asks whether or not $\delta(G) \geq k$ for a given graph G and a positive integer k. This problem is known to be NP-complete (cf. [GJ79]), and it remains NP-complete even if the given graph belongs to certain special classes of perfect graphs including chordal and bipartite graphs; see the references in Section 2. Feige et al. [FHK00] established nearly optimal approximation algorithms for the domatic number.

Expensive resources should not be wasted. Given a graph G and a positive integer i, how hard is it to determine whether or not $\delta(G)$ equals i *exactly*? More generally, given a graph G and a list $M_k = \{i_1, i_2, \ldots, i_k\}$ of k positive integers, how hard is it to determine whether or not $\delta(G)$ equals some i_j *exactly*? Motivated by such *exact versions of* NP-*complete optimization problems*, Papadimitriou and Yannakakis introduced in their seminal paper [PY84] the class DP, which consists of the differences of any two NP sets. They also studied various other important classes of problems that belong to DP, including *facet problems*, *unique solution problems*, and *critical problems*, and they proved many of them complete for DP. Cai and Meyer [CM87] showed that Minimal-3-Uncolorability is DP-complete, a critical graph problem that asks whether a given graph is not 3-colorable, but deleting any of its vertices makes it 3-colorable.

Generalizing DP, Cai et al. [CGH$^+$88,CGH$^+$89] introduced and studied $\mathrm{BH(NP)} = \bigcup_{k\geq 1} \mathrm{BH}_k\mathrm{(NP)}$, the boolean hierarchy over NP; see Section 2 for the definition. Note that DP is the second level of this hierarchy. Wagner [Wag87] identified a set of conditions sufficient to prove BH_k(NP)-hardness for each k, and he applied his sufficient conditions to prove a host of exact versions of NP-complete optimization problems complete for the levels of the boolean hierarchy. To state just one such result, Wagner [Wag87] proved that the problem of determining whether or not the chromatic number of a given graph is *exactly* one of k given values is complete for $\mathrm{BH}_{2k}\mathrm{(NP)}$. The chromatic number of a graph G, denoted by $\chi(G)$, is the minimum number of colors needed to color the vertices of G such that no two adjacent vertices receive the same color. In particular, for $k = 1$, Wagner showed that for any fixed integer $i \geq 7$, it is DP-complete to determine whether or not $\chi(G) = i$ for a given graph G. Recently, Rothe [Rot01] (see also [RSV02]) optimally strengthened Wagner's result by showing $\mathrm{BH}_{2k}(\mathrm{NP})$-completeness of the exact chromatic number problem using the smallest number of colors possible. In particular, it is DP-complete to determine whether or not $\chi(G) = 4$, yet the problem of determining whether or not $\chi(G) = 3$ is in NP and thus cannot be DP-complete unless the boolean hierarchy over NP collapses to its first level.

Wagner's technique was also useful in proving certain natural problems complete for $\mathrm{P}_{||}^{\mathrm{NP}}$, the class of problems solvable in polynomial time via parallel (i.e., truth-table) access to NP. For example, the winner problem for Carroll elections [HHR97a,HHR97b] and for Young elections [RSV02,RSV] as well as the problem of determining when certain graph heuristics work well [HR98,HRS02] each are complete for $\mathrm{P}_{||}^{\mathrm{NP}}$.

In Section 2, we prove that determining whether or not the domatic number of a given graph equals exactly one of k given values is complete for $\mathrm{BH}_{2k}(\mathrm{NP})$. In particular, for $k = 1$ and any fixed integer $i \geq 5$, it is DP-complete to determine whether or not $\delta(G) = i$ for a given graph G. In Section 3, we prove similar results for the exact conveyor flow shop problem.

2 The Exact Domatic Number Problem

We start by introducing some graph-theoretical notation. For any graph G, $V(G)$ denotes the vertex set of G, and $E(G)$ denotes the edge set of G. All graphs in this paper are undirected, simple graphs. That is, edges are unordered pairs of vertices, and there are neither multiple nor reflexive edges (i.e., for any two vertices u and v, there is at most one edge of the form $\{u, v\}$, and there is no edge of the form $\{u, u\}$). Also, all graphs considered do not have isolated vertices. For any vertex $v \in V(G)$, the *degree of* v (denoted by $\mathit{deg}_G(v)$) is the number of vertices adjacent to v in G; if G is clear from the context, we omit the subscript and simply write $\mathit{deg}(v)$. Let $\mathit{max\text{-}deg}(G) = \max_{v \in V(G)} \mathit{deg}(v)$ denote the maximum degree of the vertices of graph G, and let $\mathit{min\text{-}deg}(G) = \min_{v \in V(G)} \mathit{deg}(v)$ denote the minimum degree of the vertices of graph G.

A graph G is said to be k-*colorable* if its vertices can be colored with no more than k colors such that no two adjacent vertices receive the same color. The *chromatic number of* G, denoted by $\chi(G)$, is defined to be the smallest k such that G is k-colorable. In particular, define the decision version of the 3-colorability problem, which is one of the standard NP-complete problems (cf. [GJ79]), by:

$$\texttt{3-Colorability} = \{G \mid G \text{ is a graph with } \chi(G) \leq 3\}.$$

We now define the domatic number problem.

Definition 1 *For any graph G, a* dominating set of G *is a subset $D \subseteq V(G)$ such that for each vertex $u \in V(G) - D$, there exists a vertex $v \in D$ with $\{u, v\} \in E$. The* domatic number of G, *denoted by $\delta(G)$, is the maximum number of disjoint dominating sets. Define the decision version of the domatic number problem by:*

$$\texttt{DNP} = \{\langle G, k\rangle \mid G \textit{ is a graph and } k \textit{ is a positive integer such that } \delta(G) \geq k\}.$$

Note that $\delta(G) \leq \textit{min-deg}(G) + 1$. For general graphs and for each fixed $k \geq 3$, DNP is known to be NP-complete (cf. [GJ79]), and it remains NP-complete for circular-arc graphs [Bon85], for split graphs (thus, in particular, for chordal and co-chordal graphs) [KS94], and for bipartite graphs (thus, in particular, for comparability graphs) [KS94]. In contrast, DNP is known to be polynomial-time solvable for certain other graph classes, including strongly chordal graphs (thus, in particular, for interval graphs and path graphs) [Far84] and proper circular-arc graphs [Bon85]. For graph-theoretical notions and special graph classes not defined in this extended abstract, we refer to the monograph by Brandstädt et al. [BLS99], which is a follow-up to the classic text by Golumbic [Gol80]. Feige et al. [FHK00] show that every graph G with n vertices has a domatic partition with $(1 - o(1))(\textit{min-deg}(G) + 1)/\ln n$ sets that can be found in polynomial time, which implies a $(1 - o(1)) \ln n$ approximation algorithm for the domatic number $\delta(G)$. This is a tight bound, since they also show that, for any fixed constant $\varepsilon > 0$, the domatic number cannot be approximated within a factor of $(1 - \varepsilon) \ln n$, unless NP $\subseteq$ DTIME$(n^{\log \log n})$. Finally, Feige et al. [FHK00] give a refined algorithm that yields a domatic partition of $\Omega(\delta(G)/\ln \textit{max-deg}(G))$, which implies a $\mathcal{O}(\ln \textit{max-deg}(G))$ approximation algorithm for the domatic number $\delta(G)$. For more results on the domatic number problem, see [FHK00,KS94] and the references therein.

We assume that the reader is familiar with standard complexity-theoretic notions and notation. For more background, we refer to any standard textbook on computational complexity theory such as Papadimitriou's book [Pap94]. All completeness results in this paper are with respect to the polynomial-time many-one reducibility, denoted by $\leq_{\rm m}^{\rm p}$. For sets A and B, define $A \leq_{\rm m}^{\rm p} B$ if and only if there is a polynomial-time computable function f such that for each $x \in \Sigma^*$, $x \in A$ if and only if $f(x) \in B$. A set B is $\mathcal{C}$-hard for a complexity class $\mathcal{C}$ if and only if $A \leq_{\rm m}^{\rm p} B$ for each $A \in \mathcal{C}$. A set B is $\mathcal{C}$-complete if and only if B is $\mathcal{C}$-hard and $B \in \mathcal{C}$. To define the boolean hierarchy over NP, we use the symbols $\wedge$ and $\vee$, respectively, to denote the *complex intersection* and the *complex union* of set classes. That is, for classes $\mathcal{C}$ and $\mathcal{D}$ of sets, define

$$\mathcal{C} \wedge \mathcal{D} = \{A \cap B \mid A \in \mathcal{C} \text{ and } B \in \mathcal{D}\};$$
$$\mathcal{C} \vee \mathcal{D} = \{A \cup B \mid A \in \mathcal{C} \text{ and } B \in \mathcal{D}\}.$$

Definition 2 (Cai et al. [CGH+88]) *The* boolean hierarchy over NP *is inductively defined by:*

$$\begin{aligned} \mathrm{BH}_1(\mathrm{NP}) &= \mathrm{NP}, \\ \mathrm{BH}_2(\mathrm{NP}) &= \mathrm{NP} \wedge \mathrm{coNP}, \\ \mathrm{BH}_k(\mathrm{NP}) &= \mathrm{BH}_{k-2}(\mathrm{NP}) \vee \mathrm{BH}_2(\mathrm{NP}) \quad \textit{for } k \geq 3, \textit{ and} \\ \mathrm{BH}(\mathrm{NP}) &= \bigcup_{k \geq 1} \mathrm{BH}_k(\mathrm{NP}). \end{aligned}$$

Note that $\mathrm{DP} = \mathrm{BH}_2(\mathrm{NP})$. In his seminal paper [Wag87], Wagner provided a set of conditions sufficient to prove hardness results for the levels of the boolean hierarchy over NP and for other complexity classes, respectively. His sufficient conditions were successfully applied to classify the complexity of a variety of natural, important problems, see, e.g., [Wag87,HHR97a,HHR97b,HR98, Rot01,HRS02,RSV02,RSV]. Below, we state that one of Wagner's sufficient conditions that is relevant for this paper.

Lemma 3 (Wagner; see Thm. 5.1(3) of [Wag87]) *Let A be some* NP*-complete problem, let B be an arbitrary problem, and let $k \geq 1$ be fixed. If there exists a polynomial-time computable function f such that the equivalence*

$$||\{i \mid x_i \in A\}|| \textit{ is odd} \iff f(x_1, x_2, \dots, x_{2k}) \in B \tag{2.1}$$

is true for all strings $x_1, x_2, \dots, x_{2k} \in \Sigma^*$ satisfying that for each j with $1 \leq j < 2k$, $x_{j+1} \in A$ implies $x_j \in A$, then B is $\mathrm{BH}_{2k}(\mathrm{NP})$*-hard.*

Definition 4 *Let $M_k \subseteq \mathbb{N}$ be any set containing k noncontiguous integers. Define the exact version of the domatic number problem by:*

$$\texttt{Exact-}M_k\texttt{-DNP} = \{G \mid G \textit{ is a graph and } \delta(G) \in M_k\}.$$

In particular, for each singleton $M_1 = \{t\}$, we write $\texttt{Exact-}t\texttt{-DNP} = \{G \mid \delta(G) = t\}$.

To apply Wagner's sufficient condition from Lemma 3 in the proof of the main result of this section, Theorem 6 below, we need the following lemma due to Kaplan and Shamir [KS94] that gives a reduction from 3-`Colorability` to `DNP` with useful properties. Since Kaplan and Shamir's construction will be used explicitly in the proof of Theorem 6, we present it below.

Lemma 5 (Kaplan and Shamir [KS94]) *There exists a polynomial-time many-one reduction g from* 3-`Colorability` *to* `DNP` *with the following properties:*

$$G \in \texttt{3-Colorability} \implies \delta(g(G)) = 3; \tag{2.2}$$

$$G \notin \texttt{3-Colorability} \implies \delta(g(G)) = 2. \tag{2.3}$$

Proof. The reduction g maps any given graph G to a graph H such that the implications (2.2) and (2.3) are satisfied. Since it can be tested in polynomial time whether or not a given graph is 2-colorable, we may assume, without loss of generality, that G is not 2-colorable. Recall that we also assume that G has no isolated vertices; note that the domatic number of any graph is always at least 2 if it has no isolated vertices (cf. [GJ79]). Graph H is constructed from G by creating $||E(G)||$ new vertices, one on each edge of G, and by adding new edges such that the original vertices of G form a clique. Thus, every edge of G induces a triangle in H, and every pair of nonadjacent vertices in G is connected by an edge in H. Our construction in the proof of Theorem 6 below explicitly uses this construction and, in particular, such triangles.

Let $V(G) = \{v_1, v_2, \dots, v_n\}$. Formally, define the vertex set and the edge set of H by:

$$\begin{aligned} V(H) &= V(G) \cup \{u_{i,j} \mid \{v_i, v_j\} \in E(G)\}; \\ E(H) &= \{\{v_i, u_{i,j}\} \mid \{v_i, v_j\} \in E(G)\} \cup \{\{v_j, u_{i,j}\} \mid \{v_i, v_j\} \in E(G)\} \\ &\quad \cup \{\{v_i, v_j\} \mid 1 \leq i, j \leq n \text{ and } i \neq j\}\}. \end{aligned}$$

Since, by construction, *min-deg*$(H) = 2$ and H has no isolated vertices, the inequality $\delta(H) \leq$ *min-deg*$(H) + 1$ implies that $2 \leq \delta(H) \leq 3$.

Suppose $G \in$ 3-Colorability. Let C_1, C_2, and C_3 be the three color classes of G, i.e., $C_k = \{v_i \in V(G) \mid v_i$ is colored by color $k\}$, for $k \in \{1, 2, 3\}$. Form a partition of $V(H)$ by $\hat{C}_k = C_k \cup \{u_{i,j} \mid v_i \notin C_k$ and $v_j \notin C_k\}$, for $k \in \{1, 2, 3\}$. Since for each k, $\hat{C}_k \cap V(G) \neq \emptyset$ and $V(G)$ induces a clique in H, every $\hat{C}_k$ dominates $V(G)$ in H. Also, every triangle $\{v_i, u_{i,j}, v_j\}$ contains one element from each $\hat{C}_k$, so every $\hat{C}_k$ also dominates $\{u_{i,j} \mid \{v_i, v_j\} \in E(G)\}$ in H. Hence, $\delta(H) = 3$, which proves the implication (2.2).

Conversely, suppose $\delta(H) = 3$. Given a partition of $V(H)$ into three dominating sets, $\hat{C}_1$, $\hat{C}_2$, and $\hat{C}_3$, color the vertices in $\hat{C}_k$ by color k. Every triangle $\{v_i, u_{i,j}, v_j\}$ is 3-colored, which implies that this coloring on $V(G)$ induces a legal 3-coloring of G; so $G \in$ 3-Colorability. Hence, $\chi(G) = 3$ if and only if $\delta(H) = 3$. Since $2 \leq \delta(H) \leq 3$, the implication (2.3) follows. ∎

Next, we state the main result of this section: For each fixed set M_k containing k noncontiguous integers not smaller than $4k + 1$, Exact-M_k-DNP is complete for $\mathrm{BH}_{2k}(\mathrm{NP})$, the $2k$th level of the boolean hierarchy over NP.

Theorem 6 *For fixed $k \geq 1$, let $M_k = \{4k + 1, 4k + 3, \ldots, 6k - 1\}$. Then,* Exact-$M_k$-DNP *is* $\mathrm{BH}_{2k}(\mathrm{NP})$*-complete. In particular, for $k = 1$,* Exact-5-DNP *is* DP*-complete. In contrast,* Exact-2-DNP *is in* coNP *and thus cannot be* DP*-complete unless the boolean hierarchy over* NP *collapses.*

Proof. To show that Exact-M_k-DNP is in $\mathrm{BH}_{2k}(\mathrm{NP})$, partition the problem into k subproblems

$$\texttt{Exact-}M_k\texttt{-DNP} = \bigcup_{i \in M_k} \texttt{Exact-}i\texttt{-DNP}.$$

Every set Exact-i-DNP can be rewritten as

$$\texttt{Exact-}i\texttt{-DNP} = \{G \mid \delta(G) \geq i\} \cap \{G \mid \delta(G) < i + 1\}.$$

Clearly, the set $\{G \mid \delta(G) \geq i\}$ is in NP, and the set $\{G \mid \delta(G) < i + 1\}$ is in coNP. It follows that Exact-i-DNP is in DP, for each $i \in M_k$. By definition, Exact-M_k-DNP is in $\mathrm{BH}_{2k}(\mathrm{NP})$.

In particular, suppose $k = 1$ and consider the problem

$$\texttt{Exact-2-DNP} = \{G \mid \delta(G) \leq 2\} \cap \{G \mid \delta(G) \geq 2\}.$$

Since every graph without isolated vertices has a domatic number of at least 2 (cf. [GJ79]), the set $\{G \mid \delta(G) \geq 2\}$ is in P. On the other hand, the set $\{G \mid \delta(G) \leq 2\}$ is in coNP, so Exact-2-DNP is also in coNP and, thus, cannot be DP-complete unless the boolean hierarchy over NP collapses to its first level.

The proof that Exact-M_k-DNP is $\mathrm{BH}_{2k}(\mathrm{NP})$-hard draws on Lemma 3 with 3-Colorability being the NP-complete set A and with Exact-M_k-DNP being the set B from this lemma. Fix any $2k$ graphs $G_1, G_2, \ldots, G_{2k}$ satisfying that for each j with $1 \leq j < 2k$, if G_{j+1} is in 3-Colorability, then so is G_j. Without loss of generality, we assume that none of these graphs G_j is 2-colorable, nor does it contain isolated vertices, and we assume that $\chi(G_j) \leq 4$ for each j. Applying the Lemma 5 reduction g from 3-Colorability to DNP, we obtain $2k$ graphs $H_j = g(G_j)$, $1 \leq j \leq 2k$, each satisfying the implications (2.2) and (2.3). Hence, for each j, $\delta(H_j) \in \{2, 3\}$, and $\delta(H_{j+1}) = 3$ implies $\delta(H_j) = 3$.

We now define a polynomial-time computable function f that maps the graphs $G_1, G_2, \ldots, G_{2k}$ to a graph H such that Equation (2.1) from Lemma 3 is satisfied. The graph H is constructed from the graphs $H_1, H_2, \ldots, H_{2k}$ such that $\delta(H) = \sum_{j=1}^{2k} \delta(H_j)$. Note that the analogous property for the chromatic number (i.e., $\chi(H) = \sum_{j=1}^{2k} \chi(H_j)$) is easy to achieve by simply joining[1] the graphs H_j ([Wag87], see also [Rot01]). However, for the domatic number, the construction is more complicated. We first describe it for the special case that $k = 1$, and then explain the general case. For $k = 1$, we are given two graphs, H_1 and H_2, as above. Construct a gadget connecting H_1 and H_2 as follows. Recalling the construction from Lemma 5, let T_1 with $V(T_1) = \{v_q, u_{q,r}, v_r\}$ be any fixed triangle in H_1, and let T_2 with $V(T_2) = \{v_s, u_{s,t}, v_t\}$ be any fixed triangle in H_2. Connect T_1 and T_2 using the gadget that is shown in Figure 1. That is, add six new vertices $a_1, a_2, \ldots, a_6$, and add the following set of edges:

$$
\begin{aligned}
&\{\{v_q, a_1\}, \{v_q, a_2\}, \{v_q, a_4\}, \{v_q, a_5\}, \{v_q, a_6\},\\
&\{u_{q,r}, a_1\}, \{u_{q,r}, a_3\}, \{u_{q,r}, a_4\}, \{u_{q,r}, a_5\}, \{u_{q,r}, a_6\},\\
&\{v_r, a_2\}, \{v_r, a_3\}, \{v_r, a_4\}, \{v_r, a_5\}, \{v_r, a_6\},\\
&\{v_s, a_1\}, \{v_s, a_2\}, \{v_s, a_3\}, \{v_s, a_4\}, \{v_s, a_5\},\\
&\{u_{s,t}, a_1\}, \{u_{s,t}, a_2\}, \{u_{s,t}, a_3\}, \{u_{s,t}, a_4\}, \{u_{s,t}, a_6\},\\
&\{v_t, a_1\}, \{v_t, a_2\}, \{v_t, a_3\}, \{v_t, a_5\}, \{v_t, a_6\}\}.
\end{aligned}
$$

Using pairwise disjoint copies of the gadget from Figure 1, connect each pair of triangles from H_1 and H_2 and call the resulting graph H. Since $\mathit{deg}(a_i) = 5$ for each gadget vertex a_i, we have $\delta(H) \leq 6$, regardless of the domatic numbers of H_1 and H_2. We now show that $\delta(H) = \delta(H_1) + \delta(H_2)$.

Let $D_1, D_2, \ldots, D_{\delta(H_1)}$ be $\delta(H_1)$ pairwise disjoint sets dominating H_1, and let $D_{\delta(H_1)+1}$, $D_{\delta(H_1)+2}$, …, $D_{\delta(H_1)+\delta(H_2)}$ be $\delta(H_2)$ pairwise disjoint sets dominating H_2. Distinguish the following three cases.

Case 1: $\boldsymbol{\delta(H_1) = \delta(H_2) = 3}$. Consider any fixed D_j, where $1 \leq j \leq 3$. Since D_j dominates H_1, every triangle T_1 of H_1 has exactly one vertex in D_j. Fix T_1, and suppose $V(T_1) = \{v_q, u_{q,r}, v_r\}$ and, say, $V(T_1) \cap D_j = \{v_q\}$; the other cases are analogous. For each triangle T_2 of H_2, say T_2 with $V(T_2) = \{v_s, u_{s,t}, v_t\}$, let $a_1^{T_2}, a_2^{T_2}, \ldots, a_6^{T_2}$ be the gadget vertices connecting T_1 and T_2 as in Figure 1. Note that exactly one of these gadget vertices, $a_3^{T_2}$, is not adjacent to v_q. For each triangle T_2, add the missing gadget vertex to D_j, and define $\hat{D}_j = D_j \cup \{a_3^{T_2} \mid T_2 \text{ is a triangle of } H_2\}$. Since every vertex of H_2 is contained in some triangle T_2 of H_2 and since $a_3^{T_2}$ is adjacent to each vertex in T_2, $\hat{D}_j$ dominates H_2. Also, $\hat{D}_j \supseteq D_j$ dominates H_1, and since v_q is adjacent to each $a_i^{T_2}$ except $a_3^{T_2}$ for each triangle T_2 of H_2, $\hat{D}_j$ dominates every gadget vertex of H. Hence, $\hat{D}_j$ dominates H. By a

[1] The join operation $\bowtie$ on graphs is defined as follows: Given two disjoint graphs A and B, their join $A \bowtie B$ is the graph with vertex set $V(A \bowtie B) = V(A) \cup V(B)$ and edge set $E(A \bowtie B) = E(A) \cup E(B) \cup \{\{a, b\} \mid a \in V(A) \text{ and } b \in V(B)\}$. Note that $\bowtie$ is an associative operation on graphs and $\chi(A \bowtie B) = \chi(A) + \chi(B)$.

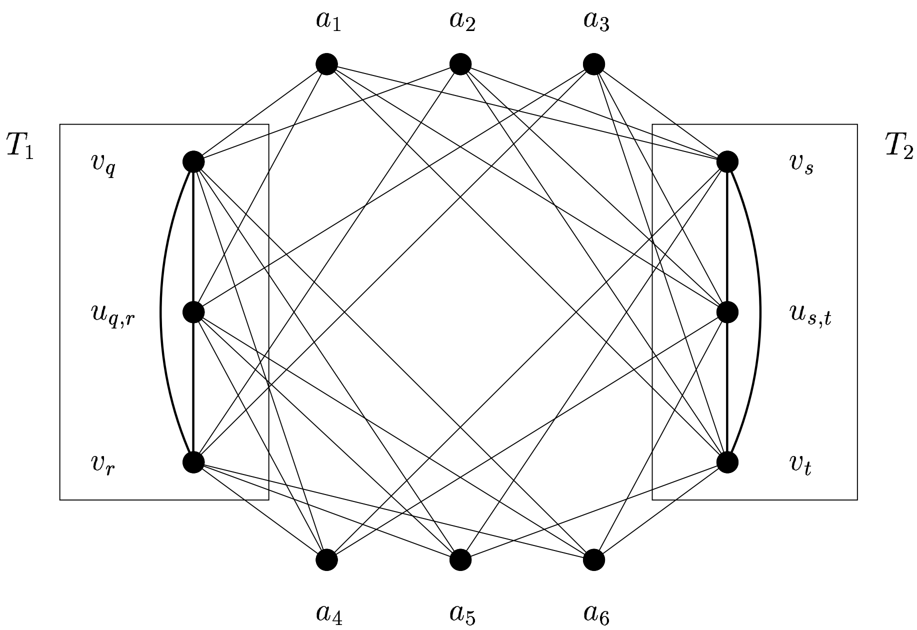

Figure 1: Gadget connecting two triangles T_1 and T_2.

symmetric argument, every set D_j, where $4 \leq j \leq 6$, dominating H_2 can be extended to a set $\hat{D}_j$ dominating the entire graph H. By contruction, the sets $\hat{D}_j$ with $1 \leq j \leq 6$ are pairwise disjoint. Hence, $\delta(H) = 6 = \delta(H_1) + \delta(H_2)$.

Case 2: $\boldsymbol{\delta(H_1) = 3}$ **and** $\boldsymbol{\delta(H_2) = 2}$. As in Case 1, we can add appropriate gadget vertices to the five given sets $D_1, D_2, \dots, D_5$ to obtain five pairwise disjoint sets $\hat{D}_1, \hat{D}_2, \dots, \hat{D}_5$ such that each $\hat{D}_i$ dominates the entire graph H. It follows that $5 \leq \delta(H) \leq 6$. It remains to show that $\delta(H) \neq 6$. For a contradiction, suppose that $\delta(H) = 6$. Look at Figure 1 showing the gadget between any two triangles T_1 and T_2 belonging to H_1 and H_2, respectively. Fix T_1 with $V(T_1) = \{v_q, u_{q,r}, v_r\}$. The only way (except for renaming the dominating sets) to partition the graph H into six dominating sets, say $E_1, E_2, \dots, E_6$, is to assign to the sets E_i the vertices of T_1, of H_2, and of the gadgets connected with T_1 as follows:

- E_1 contains v_q and $\{a_3^{T_2} \mid T_2$ is a triangle in $H_2\}$,
- E_2 contains $u_{q,r}$ and $\{a_2^{T_2} \mid T_2$ is a triangle in $H_2\}$,
- E_3 contains v_r and $\{a_1^{T_2} \mid T_2$ is a triangle in $H_2\}$,
- E_4 contains $v_s \in T_2$, for each triangle T_2 of H_2, and $\{a_6^{T_2} \mid T_2$ is a triangle in $H_2\}$,
- E_5 contains $u_{s,t} \in T_2$, for each triangle T_2 of H_2, and $\{a_5^{T_2} \mid T_2$ is a triangle in $H_2\}$,
- E_6 contains $v_t \in T_2$, for each triangle T_2 of H_2, and $\{a_4^{T_2} \mid T_2$ is a triangle in $H_2\}$.

Hence, all vertices from H_2 must be assigned to the three dominating sets E_4, E_5, and E_6, which induces a partition of H_2 into three dominating sets. This contradicts the case assumption that $\delta(H_2) = 2$. Hence, $\delta(H) = 5 = \delta(H_1) + \delta(H_2)$.

Case 3: $\boldsymbol{\delta(H_1) = \delta(H_2) = 2}$. As in the previous two cases, we can add appropriate gadget vertices to the four given sets D_1, D_2, D_3, and D_4 to obtain a partition of $V(H)$ into four sets $\hat{D}_1$, $\hat{D}_2$, $\hat{D}_3$, and $\hat{D}_4$ such that each $\hat{D}_i$ dominates the entire graph H. It follows that $4 \leq \delta(H) \leq 6$. By the same arguments as in Case 2, $\delta(H) \neq 6$. It remains to show that $\delta(H) \neq 5$. For a contradiction, suppose that $\delta(H) = 5$. Look at Figure 1 showing the gadget between any two triangles T_1 and T_2 belonging to H_1 and H_2, respectively. Suppose H is partitioned into five dominant sets $E_1, E_2, \ldots, E_5$.

First, we show that neither T_1 nor T_2 can have two vertices belonging to the same dominating set. Suppose otherwise, and let, for example, v_q and $u_{q,r}$ be both in E_1, and let v_r be in E_2; all other cases are treated analogously. This implies that the vertices v_s, $u_{s,t}$, and v_t in T_2 must be assigned to the other three dominating sets, E_3, E_4, and E_5, since otherwise one of the sets E_i would not dominate all gadget vertices a_i, $1 \leq j \leq 6$. Since T_1 is connected with each triangle of H_2 via some gadget, the same argument shows that $V(H_2)$ can be partitioned into three dominating set, which contradicts the assumption that $\delta(H_2) = 2$.

Hence, the vertices of T_1 are assigned to three different dominating sets, say E_1, E_2, and E_3. Then, every triangle T_2 of H_2 must have one of its vertices in E_4, one in E_5, and one in either one of E_1, E_2, and E_3. Again, this induces a partition of H_2 into three dominating set, which contradicts the assumption that $\delta(H_2) = 2$. It follows that $\delta(H) \neq 5$, so $\delta(H) = 4 = \delta(H_1) + \delta(H_2)$.

By construction, $\delta(H_2) = 3$ implies $\delta(H_1) = 3$, and thus the case "$\delta(H_1) = 2$ and $\delta(H_2) = 3$" cannot occur. The case distinction is complete.

Define $f(G_1, G_2) = H$. Note that f is polynomial-time computable and, by the case distinction above, f satisfies Equation (2.1):

$$\begin{aligned} & G_1 \in \texttt{3-Colorability} \text{ and } G_2 \notin \texttt{3-Colorability} \\ \iff & \delta(H_1) = 3 \text{ and } \delta(H_2) = 2 \\ \iff & \delta(H) = \delta(H_1) + \delta(H_2) = 5 \\ \iff & f(G_1, G_2) = H \in \texttt{Exact-5-DNP}. \end{aligned}$$

Applying Lemma 3 with $k = 1$, it follows that Exact-5-DNP is DP-complete.

To prove the general case, fix any $k \geq 1$. Recall that we are given the graphs $H_1, H_2, \ldots, H_{2k}$ that are constructed from $G_1, G_2, \ldots, G_{2k}$. Generalize the above construction of graph H as follows. For any fixed sequence $T_1, T_2, \ldots, T_{2k}$ of triangles, where T_i belongs to H_i, add $6k$ new gadget vertices $a_1, a_2, \ldots, a_{6k}$ and, for each i with $1 \leq i \leq 2k$, associate the three gadget vertices $a_{1+3(i-1)}$, $a_{2+3(i-1)}$, and a_{3i} with the triangle T_i. For each i with $1 \leq i \leq 2k$, connect T_i with every T_j, where $1 \leq j \leq 2k$ and $i \neq j$, via the same three gadget vertices $a_{1+3(i-1)}$, $a_{2+3(i-1)}$, and a_{3i} associated with T_i the same way T_1 and T_2 are connected in Figure 1 via the vertices a_1, a_2, and a_3. It follows that $deg(a_i) = 6k - 1$ for each i, so $\delta(H) \leq 6k$. An argument analogous to the above case distinction shows that $\delta(H) = \sum_{j=1}^{2k} \delta(H_j)$, and it follows that:

$$\begin{aligned}
&||\{i \mid G_i \in \texttt{3-Colorability}\}|| \text{ is odd} \\
&\iff (\exists i : 1 \leq i \leq k)\ [\chi(G_1) = \cdots = \chi(G_{2i-1}) = 3 \text{ and } \chi(G_{2i}) = \cdots = \chi(G_{2k}) = 4] \\
&\iff (\exists i : 1 \leq i \leq k)\ [\delta(H_1) = \cdots = \delta(H_{2i-1}) = 3 \text{ and } \delta(H_{2i}) = \cdots = \delta(H_{2k}) = 2] \\
&\iff (\exists i : 1 \leq i \leq k)\ \left[\delta(H) = \sum_{j=1}^{2k} \delta(H_j) = 3(2i-1) + 2(2k-2i+1)\right] \\
&\iff (\exists i : 1 \leq i \leq k)\ [\delta(H) = 4k + 2i - 1] \\
&\iff \delta(H) \in \{4k+1, 4k+3, \ldots, 6k-1\} \\
&\iff f(G_1, G_2, \ldots, G_{2k}) = H \in \texttt{Exact-}M_k\texttt{-DNP}.
\end{aligned}$$

Thus, f satisfies Equation (2.1). By Lemma 3, Exact-M_k-DNP is $\mathrm{BH}_{2k}(\mathrm{NP})$-complete. ∎

3 The Exact Conveyor Flow Shop Problem

The conveyor flow shop problem is a minimization problem arising in real-world applications in the wholesale business, where warehouses are supplied with goods from a central storehouse. Suppose you are given m machines, $P_1, P_2, \ldots, P_m$, and n jobs, $J_1, J_2, \ldots, J_n$. Conveyor belt systems are used to convey jobs from machine to machine at which they are to be processed in a "permutation flow shop" manner. That is, the jobs visit the machines in the fixed order $P_1, P_2, \ldots, P_m$, and the machines process the jobs in the fixed order $J_1, J_2, \ldots, J_n$. An $(n \times m)$ task matrix $\mathcal{M} = (\mu_{j,p})_{j,p}$ with $\mu_{j,p} \in \{0, 1\}$ provides the information which job has to be processed at which machine: $\mu_{j,p} = 1$ if job J_j is to be processed at machine P_p, and $\mu_{j,p} = 0$ otherwise. Every machine can process at most one job at a time. There is one worker supervising the system. Every machine can process a job only if the worker is present, which means that the worker occasionally has to move from one machine to another. If the worker is currently not present at some machine, jobs can be queued in a buffer at this machine. The objective is to minimize the movement of the worker, where we assume the "unit distance" between any two machines, i.e., to measure the worker's movement, we simply count how many times he has switched machines until the complete task matrix has been processed.[2] Let $\Delta_{\min}(\mathcal{M})$ denote the minimum number of machine switches needed for the worker to completely process a given task matrix $\mathcal{M}$, where the minimum is taken over all possible orders in which the tasks in $\mathcal{M}$ can be processed. Define the decision version of the conveyor flow shop problem by:

$$\texttt{CFSP} = \{\langle \mathcal{M}, k\rangle \mid \mathcal{M} \text{ is a task matrix and } k \text{ is a positive integer such that } \Delta_{\min}(\mathcal{M}) \leq k\}.$$

[2]In this paper, we do not consider possible generalizations of the problem CFSP such as other distance functions, variable job sequences, more than one worker, etc. We refer to [Esp01] for results on such more general problems.

Espelage and Wanke [EW00,Esp01,EW01,EW03] introduced and studied the problem CFSP, and variations thereof, extensively. We are interested in the complexity of the exact version of CFSP.

Definition 7 *Define the exact version of the conveyor flow shop problem by:*

$$\texttt{Exact-}k\texttt{-CFSP} = \left\{ \langle \mathcal{M}, S_k \rangle \;\middle|\; \begin{array}{l} \mathcal{M} \textit{ is a task matrix and } S_k \subseteq \mathbb{N} \textit{ is a set of } k \\ \textit{noncontiguous integers with } \Delta_{\min}(\mathcal{M}) \in S_k \end{array} \right\}.$$

To show that CFSP is NP-complete, Espelage [Esp01, pp. 27–44] provided, in a rather involved 17 pages proof, a reduction g from the 3-SAT problem to CFSP, via the intermediate problem of finding a "minimum valid block cover" of a given task matrix $\mathcal{M}$. In particular, finding a minimum block cover of $\mathcal{M}$ directly yields a minimum number of machine switches. Espelage's reduction can easily be modified so as to have certain useful properties, which we state in the following lemma. The details of this modification can be found in [Rie02]; in particular, prior to the Espelage reduction, a reduction from the (unrestricted) satisfiability problem to 3-SAT is used that has the properties stated as Equations (3.4) and (3.5) below.

Lemma 8 (Espelage and Riege; see pp. 27–44 of [Esp01] and pp. 37–42 of [Rie02]) *There exists a polynomial-time many-one reduction g that witnesses 3-SAT $\leq_m^p$ CFSP and satisfies, for each given boolean formula φ, the following properties:*

1. *$g(\varphi) = \langle \mathcal{M}_\varphi, z_\varphi \rangle$, where $\mathcal{M}_\varphi$ is a task matrix and $z_\varphi \in \mathbb{N}$ is an odd number.*
2. *$\Delta_{\min}(\mathcal{M}_\varphi) = z_\varphi + u_\varphi$, where u_φ denotes the minimum number of clauses of φ not satisfied under assignment t, where the minimum is taken over all assignments t of φ. Moreover, $u_\varphi = 0$ if $\varphi \in$ 3-SAT, and $u_\varphi = 1$ if $\varphi \notin$ 3-SAT.*

In particular, $\varphi \in$ 3-SAT if and only if $\Delta_{\min}(\mathcal{M}_\varphi)$ is odd.

Theorem 9 *For each $k \geq 1$, Exact-k-CFSP is $\mathrm{BH}_{2k}(\mathrm{NP})$-complete. In particular, for $k = 1$, Exact-1-CFSP is DP-complete.*

Proof. Analogously to the proof of Theorem 6, we can show that Exact-k-CFSP is in $\mathrm{BH}_{2k}(\mathrm{NP})$. To prove $\mathrm{BH}_{2k}(\mathrm{NP})$-hardness of Exact-$k$-CFSP, we again apply Lemma 3, with some fixed NP-complete problem A and with Exact-k-CFSP being the problem B from this lemma. The reduction f satisfying Equation (2.1) from Lemma 3 is defined by using two polynomial-time many-one reductions, g and h.

We now define the reductions g and h. Fix the NP-complete problem A. Let $x_1, x_2, \ldots, x_{2k}$ be strings in Σ^* satisfying that $c_A(x_1) \geq c_A(x_2) \geq \cdots \geq c_A(x_{2k})$, where c_A denotes the characteristic function of A, i.e., $c_A(x) = 1$ if $x \in A$, and $c_A(x) = 0$ if $x \notin A$. Wagner [Wag87] observed that the standard reduction (cf. [GJ79]) from the (unrestricted) satisfiability problem to 3-SAT can be easily modified so as to yield a reduction h from A to 3-SAT (via the intermediate satisfiability problem) such that, for each $x \in \Sigma^*$, the boolean formula $\varphi = h(x)$ satisfies the following properties:

$$x \in A \implies s_\varphi = m_\varphi; \tag{3.4}$$

$$x \notin A \implies s_\varphi = m_\varphi - 1, \tag{3.5}$$

where $s_\varphi = \max_t\{\ell \mid \ell$ clauses of φ are satisfied under assignment $t\}$, and m_φ denotes the number of clauses of φ. Moreover, m_φ is always odd.

Let $\varphi_1, \varphi_2, \ldots, \varphi_{2k}$ be the boolean formulas after applying reduction h to each given $x_i \in \Sigma^*$, i.e., $\varphi_i = h(x_i)$ for each i. For $i \in \{1, 2, \ldots, 2k\}$, let $m_i = m_{\varphi_i}$ be the number of clauses in φ_i, and let $s_i = s_{\varphi_i}$ denote the maximum number of satisfiable clauses of φ_i, where the maximum is taken over all assignments of φ_i. For each i, apply the Lemma 8 reduction g from 3-SAT to CFSP to obtain $2k$ pairs $\langle \mathcal{M}_i, z_i \rangle = g(\varphi_i)$, where each $\mathcal{M}_i = \mathcal{M}_{\varphi_i}$ is a task matrix and each $z_i = z_{\varphi_i}$ is the odd number corresponding to φ_i according to Lemma 8. Use these $2k$ task matrices to form a new task matrix:

$$\mathcal{M} = \begin{pmatrix} \mathcal{M}_1 & 0 & \cdots & 0 \\ 0 & \mathcal{M}_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \mathcal{M}_{2k} \end{pmatrix}.$$

Every task of some matrix $\mathcal{M}_i$, where $1 \leq i \leq 2k$, can be processed only if all tasks of the matrices $\mathcal{M}_j$ with $j < i$ have already been processed; see [Esp01,Rie02] for arguments as to why this is true. This implies that:

$$\Delta_{\min}(\mathcal{M}) = \sum_{i=1}^{2k} \Delta_{\min}(\mathcal{M}_i).$$

Let $z = \sum_{i=1}^{2k} z_i$; note that z is even. Define the set $S_k = \{z+1, z+3, \ldots, z+2k-1\}$, and define the reduction f by $f(x_1, x_2, \ldots, x_{2k}) = \langle \mathcal{M}, S_k \rangle$. Clearly, f is polynomial-time computable.

Let $u_i = u_{\varphi_i} = \min_t\{\ell \mid \ell$ clauses of φ_i are not satisfied under assignment $t\}$. Equations (3.4) and (3.5) then imply that for each i:

$$u_i = m_i - s_i = \begin{cases} 0 & \text{if } x_i \in A \\ 1 & \text{if } x_i \notin A. \end{cases}$$

Recall that, by Lemma 8, we have $\Delta_{\min}(\mathcal{M}_i) = z_i + u_i$. Hence:

$$\begin{aligned}
&||\{i \mid x_i \in A\}|| \text{ is odd} \\
\Leftrightarrow\ & (\exists i : 1 \leq i \leq k)\ [x_1, \ldots, x_{2i-1} \in A \text{ and } x_{2i}, \ldots, x_{2k} \notin A] \\
\Leftrightarrow\ & (\exists i : 1 \leq i \leq k)\ [s_1 = m_1, \ldots, s_{2i-1} = m_{2i-1} \text{ and } s_{2i} = m_{2i} - 1, \ldots, s_{2k} = m_{2k} - 1] \\
\Leftrightarrow\ & (\exists i : 1 \leq i \leq k)\ [\Delta_{\min}(\mathcal{M}_1) = z_1, \ldots, \Delta_{\min}(\mathcal{M}_{2i-1}) = z_{2i-1} \text{ and} \\
& \qquad \Delta_{\min}(\mathcal{M}_{2i}) = z_{2i} + 1, \ldots, \Delta_{\min}(\mathcal{M}_{2k}) = z_{2k} + 1] \\
\Leftrightarrow\ & (\exists i : 1 \leq i \leq k)\ \left[\Delta_{\min}(\mathcal{M}) = \sum_{j=1}^{2k} \Delta_{\min}(\mathcal{M}_j) = \left(\sum_{j=1}^{2k} z_j\right) + 2k - 2i + 1\right] \\
\Leftrightarrow\ & \Delta_{\min}(\mathcal{M}) \in S_k = \{z+1, z+3, \ldots, z+2k-1\} \\
\Leftrightarrow\ & f(x_1, x_2, \ldots, x_{2k}) = \langle \mathcal{M}, S_k \rangle \in \texttt{Exact-}k\texttt{-CFSP}.
\end{aligned}$$

Thus, f satisfies Equation (2.1). By Lemma 3, Exact-k-CFSP is $\mathrm{BH}_{2k}(\mathrm{NP})$-complete. ∎

4 Conclusions and Open Questions

In this paper, we have shown that the exact versions of the domatic number problem and of the conveyor flow shop problem are complete for the levels of the boolean hierarchy over NP. In particular, for $k = 1$ and for each given integer $i \geq 5$, it is DP-complete to determine whether or not $\delta(G) = i$ for a given graph G. In contrast, Exact-2-DNP is in coNP, and thus this problem cannot be DP-complete unless the boolean hierarchy collapses. For $i \in \{3, 4\}$, the question of whether or not the problems Exact-i-DNP are DP-complete remains an interesting open problem. As mentioned in the introduction, the corresponding gap for the exact chromatic number problem was recently closed by Rothe [Rot01]; see also [RSV02]. His reduction uses both the standard reduction from 3-SAT to 3-Colorability (cf. [GJ79]) and a very clever reduction found by Guruswami and Khanna [GK00]. The decisive property of the Guruswami–Khanna reduction is that it maps each satisfiable formula φ to a graph G with $\chi(G) = 3$, and it maps each unsatisfiable formula φ to a graph G with $\chi(G) = 5$. That is, the graphs they construct are never 4-colorable. To close the above-mentioned gap for the exact domatic number problem, one would have to find a reduction from some NP-complete problem to DNP with a similarly strong property: the reduction would have to yield graphs that never have a domatic number of 3.

Note that in defining the exact conveyor flow shop problem, we do not specify a fixed set S_k with k fixed values as problem parameters; see Definition 7. Rather, only the cardinality k of such sets is given as a parameter, and S_k is part of the problem instance of Exact-k-CFSP. The reason is that the actual values of S_k depend on the input of the reduction f defined in the proof of Theorem 9. In particular, the number z_φ from Lemma 8, which is used to define the number $z = \sum_{i=1}^{2k} z_i$ in the proof of Theorem 9, has the following form (see [Esp01,Rie02]):

$$z_\varphi = 28n_K + 27n_{\overline{K}} + 8n_U + 90mt + 99m,$$

where t is the number of variables and m is the number of clauses of the given boolean formula φ, and n_K, $n_{\overline{K}}$, and n_U denote respectively the number of “coupling, inverting coupling, and interrupting elements” of the “minimum valid block cover” constructed in the Espelage reduction [Esp01] from 3-SAT to CFSP. It would be interesting to know whether one can obtain $\mathrm{BH}_{2k}(\mathrm{NP})$-completeness of Exact-$k$-CFSP even if a set S_k of k fixed values is specified a priori.

References

[BLS99] A. Brandstädt, V. Le, and J. Spinrad. *Graph Classes: A Survey*. SIAM Monographs on Discrete Mathematics and Applications. Society for Industrial and Applied Mathematics, Philadelphia, PA, 1999.

[Bon85] M. Bonuccelli. Dominating sets and dominating number of circular arc graphs. *Discrete Applied Mathematics*, 12:203–213, 1985.

[CGH+88] J. Cai, T. Gundermann, J. Hartmanis, L. Hemachandra, V. Sewelson, K. Wagner, and G. Wechsung. The boolean hierarchy I: Structural properties. *SIAM Journal on Computing*, 17(6):1232–1252, 1988.

[CGH+89] J. Cai, T. Gundermann, J. Hartmanis, L. Hemachandra, V. Sewelson, K. Wagner, and G. Wechsung. The boolean hierarchy II: Applications. *SIAM Journal on Computing*, 18(1):95–111, 1989.

[CM87] J. Cai and G. Meyer. Graph minimal uncolorability is D^{P}-complete. *SIAM Journal on Computing*, 16(2):259–277, April 1987.

[Esp01] W. Espelage. *Bewegungsminimierung in der Förderband-Flow-Shop-Verarbeitung*. PhD thesis, Heinrich-Heine-Universität Düsseldorf, Düsseldorf, Germany, 2001. In German.

[EW00] W. Espelage and E. Wanke. Movement optimization in flow shop processing with buffers. *Mathematical Methods of Operations Research*, 51(3):495–513, 2000.

[EW01] W. Espelage and E. Wanke. A 3-approximation algorithmus for movement minimization in conveyor flow shop processing. In *Proceedings of the 26th International Symposium on Mathematical Foundations of Computer Science*, pages 363–374. Springer-Verlag *Lecture Notes in Computer Science #2136*, 2001.

[EW03] W. Espelage and E. Wanke. Movement minimization for unit distances in conveyor flow shop processing. *Mathematical Methods of Operations Research*, 57(2), 2003. To appear.

[Far84] M. Farber. Domination, independent domination, and duality in strongly chordal graphs. *Discrete Applied Mathematics*, 7:115–130, 1984.

[FHK00] U. Feige, M. Halldórsson, and G. Kortsarz. Approximating the domatic number. In *Proceedings of the 32nd ACM Symposium on Theory of Computing*, pages 134–143. ACM Press, May 2000.

[GJ79] M. Garey and D. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, New York, 1979.

[GK00] V. Guruswami and S. Khanna. On the hardness of 4-coloring a 3-colorable graph. In *Proceedings of the 15th Annual IEEE Conference on Computational Complexity*, pages 188–197. IEEE Computer Society Press, May 2000.

[Gol80] M. Golumbic. *Algorithmic Graph Theory and Perfect Graphs*. Academic Press, 1980.

[HHR97a] E. Hemaspaandra, L. Hemaspaandra, and J. Rothe. Exact analysis of Dodgson elections: Lewis Carroll's 1876 voting system is complete for parallel access to NP. *Journal of the ACM*, 44(6):806–825, November 1997.

[HHR97b] E. Hemaspaandra, L. Hemaspaandra, and J. Rothe. Raising NP lower bounds to parallel NP lower bounds. *SIGACT News*, 28(2):2–13, June 1997.

[HR98] E. Hemaspaandra and J. Rothe. Recognizing when greed can approximate maximum independent sets is complete for parallel access to NP. *Information Processing Letters*, 65(3):151–156, February 1998.

[HRS02] E. Hemaspaandra, J. Rothe, and H. Spakowski. Recognizing when heuristics can approximate minimum vertex covers is complete for parallel access to NP. In *Proceedings of the 28th International Workshop on Graph-Theoretic Concepts in Computer Science (WG 2002)*. Springer-Verlag *Lecture Notes in Computer Science #2573*, June 2002. To appear. Technical Report cs.CC/0110025 available on-line from Computing Research Repository (CoRR) at http://xxx.lanl.gov/abs/cs.CC/0110025.

[KS94] H. Kaplan and R. Shamir. The domatic number problem on some perfect graph families. *Information Processing Letters*, 49(1):51–56, January 1994.

[Pap94] C. Papadimitriou. *Computational Complexity*. Addison-Wesley, 1994.

[PY84] C. Papadimitriou and M. Yannakakis. The complexity of facets (and some facets of complexity). *Journal of Computer and System Sciences*, 28(2):244–259, 1984.

[Rie02] T. Riege. Vollständige Probleme in der Booleschen Hierarchie über NP. *Diploma thesis*, Heinrich-Heine-Universität Düsseldorf, Institut für Informatik, Düsseldorf, Germany, August 2002. In German.

[Rot01] J. Rothe. Exact complexity of Exact-Four-Colorability. Technical Report cs.CC/0109018, Computing Research Repository (CoRR), September 2001. 5 pages. Available on-line at http://xxx.lanl.gov/abs/cs.CC/0109018.

[RSV] J. Rothe, H. Spakowski, and J. Vogel. Exact complexity of the winner problem for Young elections. *Theory of Computing Systems*. To appear. A preliminary version is available on-line at http://xxx.lanl.gov/abs/cs.CC/0112021.

[RSV02] J. Rothe, H. Spakowski, and J. Vogel. Exact complexity of Exact-Four-Colorability and of the winner problem for Young elections. In R. Baeza-Yates, U. Montanari, and N. Santoro, editors, *Foundations of Information Technology in the Era of Network and Mobile Computing*, pages 310–322. Kluwer Academic Publishers, August 2002. Proceedings of the 2nd IFIP International Conference on Theoretical Computer Science, Stream 1 of the 17th IFIP World Computer Congress.

[Wag87] K. Wagner. More complicated questions about maxima and minima, and some closures of NP. *Theoretical Computer Science*, 51:53–80, 1987.

ECCC ISSN 1433-8092
http://www.eccc.uni-trier.de/eccc
ftp://ftp.eccc.uni-trier.de/pub/eccc
ftpmail@ftp.eccc.uni-trier.de, subject 'help eccc'