Electronic Colloquium on Computational Complexity, Report No. 16 (2005)

Classification of Bipartite Boolean Constraint Satisfaction through Delta-Matroid Intersection

Tomás Feder
268 Waverley St., Palo Alto, CA 94301, USA
tomas@theory.stanford.edu,
Daniel Ford
University of California Santa Cruz
ford@soe.ucsc.edu

March 27, 2004

Abstract

Matroid intersection has a known polynomial time algorithm using an oracle. We generalize this result to delta-matroids that do not have equality as a restriction, and give a polynomial time algorithm for delta-matroid intersection on delta-matroids without equality using an oracle. We note that when equality is present, delta-matroid intersection is as general as delta-matroid parity. We also obtain algorithms using an oracle for delta-matroid parity on delta-matroids without inequality, and for delta-matroid intersection where one delta-matroid does not contain either equality or inequality, and the second delta-matroid is arbitrary. Both of these results also generalize matroid intersection. The results imply a dichotomy for bipartite Boolean constraint satisfaction problems using an oracle when one of the two sides does not contain equality, leaving open cases of delta-matroid parity when both sides have equality; the results also imply a full dichotomy for k-partite Boolean constraint satisfaction problems for $k \geq 3$. We then discuss polynomial cases of Boolean constraint satisfaction problems with two occurrences per variable through delta-matroid parity that cannot be obtained using the oracle approach.

1 Introduction

An instance of the Boolean constraint satisfaction problem consists of a collection of variables ranging over the Boolean domain and a collection of constraints on them. The aim is to assign value 0 or 1 to each variable so as to satisfy all the constraints. The Boolean constraint satisfaction problem is NP-complete. Schaefer [11] considered the restriction of Boolean constraint satisfaction problems to the case where the constraints used must each belong to a given collection of allowed constraint types. Schaefer then classified the Boolean constraint satisfaction problems as polynomial time solvable or NP-complete, depending on the choice of the collection of allowed constraint types. In the case where restricting a variable to take value 0 or to take value 1 is an allowed constraint, the Schaefer polynomial cases are conjunctions (1) of Horn clauses, (2) of dual-Horn clauses, (3) of 2-satisfiability clauses, and (4) of linear equations modulo 2.

The constraint satisfaction problem with a collection of allowed constraint types can be further restricted so that each variable is only allowed to participate in two constraints. While the polynomial cases of Schaefer's classification remain polynomial under this restriction, some of the NP-complete cases may become polynomial time solvable. Feder [5] showed that the NP-complete

ISSN 1433-8092

cases of Schaefer's classification remain NP-complete unless each allowed constraint type is a delta-matroid. In that case, the problem with two occurrences for each variable is the well-known delta-matroid parity problem [1], which generalizes matroid parity [10]. Only certain families of matroid and delta-matroid parity problems are known to be polynomial time solvable. The best known such problem is graph matching.

A further restriction of Boolean constraint satisfaction problems with two occurrences per variable requires the constraints in an instance to be partitioned into two sets, so that each variable participates in only one constraint from each set. This restricted problem is known as the bipartite Boolean constraint satisfaction problem. Again, Feder [5] showed that for the bipartite Boolean constraint satisfaction problem, the NP-complete cases of Schaefer's classification remain NP-complete unless each allowed constraint type is a delta-matroid. In that case, the bipartite constraint satisfaction problem is delta-matroid intersection, which generalizes matroid intersection, and in particular bipartite graph matching.

Since matroid intersection is polynomial time solvable by the algorithm of Edmonds [4], it is natural to ask wether delta-matroid intersection is polynomial time solvable. The main difficulty is that if the equality constraint is among the allowed constraint types, matroid intersection becomes as hard as matroid parity. In fact, a bipartite Boolean constraint satisfaction problem is more restrictive than the general Boolean constraint satisfaction problem with two occurrences per variable only if the equality constraint is not among the allowed constraints. In this paper, we thus consider delta-matroid intersection in the case where the delta-matroids do not contain the equality constraint as a restriction, and give a polynomial time algorithm for the problem. This completes our first classification result for bipartite Boolean constraint satisfaction problems, which are assumed not to contain the equality constraint as an allowed constraint type, as polynomial time solvable or NP-complete.

In the model adopted, we impose no restriction on the size of constraints describing the two delta-matroids without equality to be intersected. We thus adopt the most general model, in which each of the delta-matroids is given by an oracle that can be queried in polynomial time to obtain a feasible assignment for the delta-matroid, or to determine whether a given assignment is feasible for the delta-matroid. We observe also that Schaefer's polynomial cases also remain polynomial with a slightly more powerful oracle, which allows querying the oracle to determine whether a given partial assignment can be extended to a full assignment satisfying a given constraint. Both oracles have the same power in the case of delta-matroids.

In this general oracle model, we also show that delta-matroid parity for delta-matroids that do not have the inequality constraint as a restriction can also be solved in polynomial time. We further show that intersecting a delta-matroid that has neither the equality constraint nor the inequality constraint as a restriction, with an arbitrary delta-matroid, also has a polynomial time algorithm. As matroid intersection can be represented as the intersection of two delta-matroids that contain neither the equality constraint nor the inequality constraint as a restriction, all of these results generalize matroid intersection. In fact all three results follow from a single more general algorithm for a class of delta-matroid parity problems.

This last result is then used to obtain a more general classification result for bipartite Boolean constraint satisfaction, in which the allowed constraint types may be different for both sides of the bipartition, and it is assumed that at least one side does not contain equality. If both sides contain equality, then both sides can be assumed to be the same, where the problems not yet classified are delta-matroid parity problems. We note that the polynomial cases in the classification are polynomial in the oracle model as well. See Table 1 for the classification. This also implies a dichotomy for k-partite Boolean constraint satisfaction with $k \geq 3$, where we have k sets of allowed constraint types and each variable is only allowed to participate in one constraint from each of k

- bipartite Boolean constraint satisfaction
  1. NP-complete cases
  2. Schaefer derived cases
     (a) Horn
     (b) dual-Horn
     (c) linear
     (d) 2-SAT
     (e) one side has only monadic constraints
     (f) upward 2-SAT in one side and constraints with 2-SAT downward closure in other side (and case interchanging upward and downward)
  3. one side has 2-SAT upward closure with delta-matroid downward closure and other side has 2-SAT downward closure with delta-matroid upward closure, each side is intersection of upward and downward closure, and a flat of the delta-matroid can intersect a 2-SAT clause in exactly one element only if the flat or the 2-SAT clause has only one element.
  4. delta-matroid derived cases
     (a) delta-matroid intersection without equality
     (b) delta-matroid intersection having one side without equality or inequality
     (c) upward delta-matroid in one side and constraints with delta-matroid downward closure in other side (and case interchanging upward and downward)
     (d) delta-matroid parity with equality
        i. local even or odd delta-matroid
        ii. A-local-zebra delta-matroid
        iii. linear-zebra delta-matroid
        iv. delta-matroid without inequality
        v. open cases
- k-partite Boolean constraint satisfaction for $k \geq 3$
  1. NP-complete cases
  2. polynomial cases using an oracle

Table 1: Classification of bipartite Boolean constraint satisfaction problems: cases other than zebra are also polynomial with oracle

sets of constraints of the corresponding types.

The study is thus conducted in the full generality of the oracle model. On the other hand, the general case with two occurrences per variable cannot be solved in the general oracle model. In particular, matroid parity has an exponential lower bound due to Lovász [9] in the oracle model. We thus seek to study cases of delta-matroid parity where each of the constraints used is described explicitly. In this model, Feder [5] showed that coindependent delta-matroids have a polynomial time algorithm for delta-matroid parity. We show here that coindependent delta-matroid parity has an exponential lower bound when the coindependent delta-matroid is described by an oracle. We also introduce zebra delta-matroids, as a common generalization of coindependent delta-matroids and the delta-matroids from the general factor problem that was solved by Cornuejols [2]. We show that zebra delta-matroid parity can be solved in polynomial time when each of the zebra constraints is described explicitly. We also show how to recognize certain delta-matroids that can be represented through zebra delta-matroids, thus obtaining the class of zebra-compact delta-matroids generalizing the compact delta-matroids of Istrate [8] based on the general factor problem. Finally, for any class of even delta-matroids that has a polynomial time algorithm for delta-matroid parity, such as linear matroids with the algorithms of Lovász [9], Gabow and Stallman [6], linear delta-matroids with the algorithm of Geelen, Iwata and Murota [7], or local delta-matroids with the algorithm of Dalmau and Ford [3], we define an associated class of delta-matroids that are not necessarily even, along the same line that defined zebra delta-matroids from certain even delta-matroids that can be obtained via graph matching. We show that these zebra-like delta-matroids associated with the given class of even delta-matroids also have a polynomial time algorithm for delta-matroid parity when the constraints used are described explicitly.

2 Definitions

A *delta-matroid* is a pair $M = (E, \mathcal{F})$, where E is a set and $\mathcal{F}$ is a set of subsets of E, satisfying the following axiom: For all $A, B \in \mathcal{F}$, and for all $x \in A \Delta B$, there exists a $y \in A \Delta B$ such that $A \Delta \{x, y\} \in \mathcal{F}$. Note that we may have $y = x$. The sets $A \in \mathcal{F}$ are called the *feasible sets* of the delta-matroid M.

A *restriction* of a delta-matroid $M = (E, \mathcal{F})$ is a delta-matroid $M_1 = (E_1, \mathcal{F}_1)$ with $E_1 \subseteq E$ such that for some $E_1' \subseteq E \setminus E_1$, we have $A \in \mathcal{F}_1$ if and only if $A \cup E_1' \in \mathcal{F}$. Given two delta-matroids $M_1 = (E_1, \mathcal{F}_1)$ and $M_2 = (E_2, \mathcal{F}_2)$ with $E_1 \cap E_2 = \emptyset$, the *direct sum* of M_1 and M_2 is the delta-matroid $M = (E, \mathcal{F})$ with $E = E_1 \cup E_2$ such that $A \in \mathcal{F}$ if and only if $A \cap E_1 \in \mathcal{F}_1$ and $A \cap E_2 \in \mathcal{F}_2$.

Let $M = (E, \mathcal{F})$ be a delta-matroid and $\mathcal{L}$ a partition of E into pairs. For every $u \in E$, its *mate* will be denoted by $\overline{u}$, that is $\overline{u}$ is the only element in E such that $\{u, \overline{u}\} \in \mathcal{L}$.

Let $F \in \mathcal{F}$ be a feasible set. We will let $\mathcal{L}_F$ denote the subset of $\mathcal{L}$ containing those pairs $\{u, \overline{u}\} \in \mathcal{L}$ such that either both u and $\overline{u}$ are in F or neither u nor $\overline{u}$ is in F.

An instance of the *delta-matroid parity* problem consists of a delta-matroid $M = (E, \mathcal{F})$ and a partition $\mathcal{L}$ of E into pairs. The goal is to find a feasible set $F \in \mathcal{F}$ such that $\mathcal{L}_F$ is maximum, that is, at least as large as $\mathcal{L}_G$ for any other $G \in \mathcal{F}$.

The *delta-matroid intersection* problem is the special case of the delta-matroid parity problem where $M = (E, \mathcal{F})$ is the direct sum of $M_1 = (E_1, \mathcal{F}_1)$ and $M_2 = (E_2, \mathcal{F}_2)$, and every pair in $\mathcal{L}$ contains one element in E_1 and one element in E_2.

We consider two particular delta-matroids, the *equal* delta-matroid $M_= = (\{a, b\}, \{\emptyset, \{a, b\}\})$, and the *not-equal* delta-matroid $M_{\neq} = (\{a, b\}, \{\{a\}, \{b\}\})$. Note that every delta-matroid parity problem with $M, \mathcal{L}$ is equivalent to a delta-matroid intersection problem with $M', \mathcal{L}'$, where

$M_1 = M$ and M_2 is the direct sum of $M_=$ delta-matroids, one for each pair in $\mathcal{L}$, where $\mathcal{L}'$ has the corresponding pairs $\{u, a\}$ and $\{\overline{u}, b\}$.

Thus delta-matroid intersection is a strict special case of delta-matroid parity only for delta-matroids M that do not have $M_=$ as a restriction. In an instance of delta-matroid parity or intersection, we are given an *oracle* for $M = (E, \mathcal{F})$ that can be queried to provide a particular feasible set in $\mathcal{F}$, and tested with some $A \subseteq E$ so that the oracle responds whether $A \in \mathcal{F}$, that is, whether A is feasible.

The following is known [5]. If $M = (E, \mathcal{F})$, $\mathcal{L}$ is an instance of delta-matroid parity, and $\mathcal{K}$ is a subset of pairs from $\mathcal{L}$, then we obtain a delta-matroid $M' = (E', \mathcal{F}')$ with E' consisting of the elements of E that are not in pairs in $\mathcal{K}$, and including in $\mathcal{F}'$ all sets $A \subseteq E'$ such that there exists a $B \in \mathcal{F}$ such that $B \cap E' = A$ and B is a feasible set for the delta-matroid M satisfying the pairings in $\mathcal{K}$, that is, $\mathcal{K}_B = \mathcal{K}$. We say that M' is the delta-matroid obtained from M, $\mathcal{L}$ by *contracting* $\mathcal{K}$. We can then let $\mathcal{L}' = \mathcal{L} \setminus \mathcal{K}$.

We give a polynomial time algorithm for any instance of delta-matroid parity on a delta-matroid M with pairing $\mathcal{L}$ such that no subset $\mathcal{K} \subset \mathcal{L}$ and pair $\{a, b\} \in \mathcal{L} \setminus \mathcal{K}$ are such that the delta-matroid M' obtained from M by contracting $\mathcal{K}$ has the not-equal delta-matroid $M_{\neq}$ on $\{a, b\}$ as a restriction. The bipartite Boolean constraint satisfaction classification will follow from this result.

3 Small Delta-Matroids

In this section we establish simple properties of delta-matroids with three or four elements.

Lemma 3.1 *Let $M = (\{a, b, c\}, \mathcal{F})$ be a delta-matroid with a feasible set F such that $F\Delta\{a, b, c\}$ is also feasible. Then one of $F\Delta\{a\}$, $F\Delta\{c\}$, $F\Delta\{a, c\}$ is also feasible.*

Proof. Let $A = F\Delta\{a, b, c\}$, $B = F$, and $x = b$ in the definition of delta-matroid. □

Lemma 3.2 *Let $M = (\{a, b, c\}, \mathcal{F})$ be a delta-matroid having feasible sets $F\Delta\{c\}$ and $F\Delta\{a, b\}$. Then one of of $F\Delta\{a\}$, $F\Delta\{a, c\}$, $F\Delta\{a, b, c\}$ is also feasible.*

Proof. Let $A = F\Delta\{c\}$, $B = F\Delta\{a, b\}$, and $x = a$ in the definition of delta-matroid. □

Lemma 3.3 *Let $M = (\{a, b, c, d\}, \mathcal{F})$ be a delta-matroid with a feasible set F such that $F\Delta\{a, b\}$ and $F\Delta\{a, b, c, d\}$ are also feasible. Then one of $F\Delta\{a\}$, $F\Delta\{a, c\}$, $F\Delta\{a, d\}$, $F\Delta\{c, d\}$ is also feasible.*

Proof. Consider $C = F\Delta\{a, c, d\}$. If C is feasible, let $A = F\Delta\{a, b\}$, $B = C$, and $x = b$ in the definition of delta-matroid. If C is not feasible, take $A = F\Delta\{a, b, c, d\}$, $B = F$, and $x = b$ in the definition of delta-matroid. □

Lemma 3.4 *Let $M = (\{a, b, c, d\}, \mathcal{F})$ be a delta-matroid with a feasible set F such that $F\Delta\{a, b\}$ and $F\Delta\{c, d\}$ are also feasible. Then one of $F\Delta\{a\}$, $F\Delta\{a, c\}$, $F\Delta\{a, d\}$, $F\Delta\{a, b, c, d\}$ is also feasible.*

Proof. Consider $C = F\Delta\{a, c, d\}$. If C is feasible, let $A = F\Delta\{a, b\}$, $B = C$, and $x = b$ in the definition of delta-matroid. If C is not feasible, take $A = F\Delta\{c, d\}$, $B = F\Delta\{a, b\}$, and $x = a$ in the definition of delta-matroid. □

4 Structure and Algorithm for Augmenting Paths and Blossoms

Let $M = (E, \mathcal{F})$, $\mathcal{L}$ be an instance of the delta-matroid parity problem. A *path* in M is an ordered collection $u_1, \ldots, u_n$ of different elements in E. Let $L \subseteq \mathcal{L}$ be any collection of pairs of $\mathcal{L}$. A path $u_1, \ldots, u_n$ is called *L-alternating* if: (1) for every $1 \leq 2j < n$, $\{u_{2j}, u_{2j+1}\} \in L$, (2) $\{u_1, \overline{u_1}\} \notin L$, and (3) if n is even then $\{u_1, u_n\}, \{u_n, \overline{u_n}\} \notin L$, $u_n \neq \overline{u_1}$. Let $F \in \mathcal{F}$ be a feasible set. We say that a path $u_1, \ldots, u_n$ is an *F-augmenting path* (or simply an augmenting path when F is implicit) if: (1) $F\Delta\{u_1, \ldots, u_{2j}\} \in \mathcal{F}$ for all $1 < 2j \leq n$ and (2) $F\Delta\{u_1, \ldots, u_n\} \in \mathcal{F}$.

The basic intuition behind this definition is that if F is a feasible set such that $|\mathcal{L}_F|$ is not maximum, then there exists some F-augmenting $\mathcal{L}_F$-alternating path. This path can be used to obtain a new feasible set $G = F\Delta\{u_1, \ldots, u_n\}$ which increases the objective function that we intend to maximize, $|\mathcal{L}_G| > |\mathcal{L}_F|$. In fact if $|\mathcal{L}_F|$ is not maximum, then there exists an F-augmenting $\mathcal{L}_F$-alternating path that can be computed in time polynomial in $|E|$ given a $G \in \mathcal{F}$ with $|\mathcal{L}_G| > |\mathcal{L}_F|$, see e.g. [3].

Given a feasible $F \in \mathcal{F}$, an *edge* is a pair $\{u, v\}$ of distinct elements in E such that $F\Delta\{u, v\} \in \mathcal{F}$, and a *special element* is a single element u in E such that $F\Delta\{u\} \in \mathcal{F}$.

Theorem 4.1 *Suppose M has an F-augmenting $\mathcal{L}_F$-alternating path. Let $u_1, \ldots, u_s$ be a shortest such path. Then either (1) there exists an $\mathcal{L}_F$-alternating path $v_1, \ldots, v_n$ with $v_1 = u_1$, $v_n = u_s$, such that for $2 \leq 2i \leq n$, $\{v_{2i-1}, v_{2i}\}$ is an edge, and v_n is a special element if n is odd, with each v_i among the u_j, or (2) there exists an $\mathcal{L}_F$-alternating path $w_1, \ldots, w_k$ with $w_1 = u_1$ and k odd, and a $2 \leq 2r < k$ such that for every $2 \leq 2j < k$, $\{w_{2j-1}, w_{2j}\}$ is an edge, and $\{w_k, w_{2r-1}\}$ is also an edge, with each w_i among the u_j.*

The alternating path in case (2) is called a *blossom*.

Proof. Let $u_1, \ldots, u_n$ be a shortest augmenting path. We show that either (1) for each $2 \leq 2j \leq n$ there is an edge $\{u_{2i-1}, u_{2j}\}$ for some $2 \leq 2i \leq 2j$, and if n is odd then u_n is a special element, or (2) for some $2 \leq 2k < n$ there is an edge $\{u_{2l+1}, u_{2k+1}\}$ for some $0 \leq 2l < 2k$, and for each $2 \leq 2j \leq 2k$ there is an edge $\{u_{2i-1}, u_{2j}\}$ for some $2 \leq 2i \leq 2j$. In case (1), tracing back the edges from u_{2j} for $j = n$ or $j = n-1$ to u_{2i-1}, to the mate u_{2i-2}, then the edge joining u_{2i-2} to some $u_{2i'-1}$ with $2i' < 2i$, then to the mate $u_{2i'-1}$, and so on until $u_1 = u$ is reached, gives an alternating path from u_1 to u_n that alternates going to a mate and traversing an edge. In case (2), we get such a path from u_1 to u_{2k} to the mate u_{2k+1} with an edge to u_{2l+1}, with a similar alternating path from u_{2l+1} back to u_1, which at the point it meets the path from u_1 to u_{2k} completes the blossom.

For each $2j \leq n$ and each $0 \leq 2s < 2j$, we show that either there is a $2s < 2i \leq 2j$ such that $F\Delta\{u_1, \ldots, u_{2s}, u_{2i-1}, u_{2j}\}$ is feasible, or there is a $2s < 2i < 2j$ such that $F\Delta\{u_1, \ldots, u_{2s}, u_{2i-1}, u_{2j-1}\}$ is feasible, unless there is a blossom among elements $u_1, \ldots, u_{2s}, u_{2i-1}$. The proof is by induction with decreasing s. When we reach $s = 0$, we have either the edge $\{u_{2i-1}, u_{2j}\}$ or the edge $\{u_{2i-1}, u_{2j-1}\}$ as required.

The base case $2s = 2j-2$ is verified since $F\Delta\{u_1, \ldots, u_{2j-2}, u_{2j-1}, u_{2j}\}$ is feasible by the definition of an augmenting path. Suppose the claim holds for $2s+2$, and $F\Delta\{u_1, \ldots, u_{2s}, u_{2s+1}, u_{2s+2}, u_{2i-1}, u_t\}$ is feasible, where $t = 2j$ or $t = 2j-1$. Let $G = F\Delta\{u_1, \ldots, u_{2s}\}$, and apply Lemma 3.3 with $a = u_{2s+1}$, $b = u_{2s+2}$, $c = u_{2i-1}$, $d = u_t$. We have that G, $G\Delta\{a, b\}$, and $G\Delta\{a, b, c, d\}$ are feasible. If $G\Delta\{a\}$ is feasible, we have a shorter augmenting path obtained from $G\Delta\{a\} = F\Delta\{u_1, \ldots, u_{2s}, u_{2s+1}\}$, contrary to assumption. If $G\Delta\{a, c\}$ is feasible, we have $G\Delta\{a, c\} = F\Delta\{u_1, \ldots, u_{2s}, u_{2s+1}, u_{2i-1}\}$ feasible, which inductively will give an edge $\{u_{2l+1}, u_{2k+1}\}$ as above with $2k+1 = 2i-1$ ant $2l+1 \leq 2s+1$, and thus a blossom. If $G\Delta\{a, d\}$ is feasible, we have $G\Delta\{a, d\} = F\Delta\{u_1, \ldots, u_{2s}, u_{2s+1}, u_t\}$ feasible which inductively will give an edge $\{u_{2j-1}, u_t\}$ as above with $2j-1 \leq 2s+1$. If $G\Delta\{c,d\}$ is feasible, we have $G\Delta\{c,d\} = F\Delta\{u_1,\dots,u_{2s},u_{2i-1},u_t\}$ feasible which inductively will give an edge $\{u_{2r-1},u_t\}$ as above with $2r-1 \leq 2i-1$.

It remains to show that u_n is a special element if n is odd. We show for each $0 \leq 2s < n$ that $F\Delta\{u_1,\dots,u_{2s},u_n\}$ is feasible inductively with s decreasing, unless there is a blossom. When we reach $s=0$ we have u_n as a special element. The base case $2s = n-1$ is verified since $F\Delta\{u_1,\dots,u_n\}$ is feasible. Suppose the claim holds for $2s+2$, and $F\Delta\{u_1,\dots,u_{2s},u_{2s+1},u_{2s+2},u_n\}$ is feasible. Let $G = F\Delta\{u_1,\dots,u_{2s}\}$ and apply Lemma 3.1 with $a=u_{2s}$, $b=u_{2s+1}$, $c=u_n$. We have that G and $G\Delta\{a,b,c\}$ are feasible. If $G\Delta\{a\}$ is feasible, we have a shorter augmenting path obtained from $G\Delta\{a\} = F\Delta\{u_1,\dots,u_{2s},u_{2s+1}\}$, contrary to assumption. If $G\Delta\{c\}$ is feasible, we have $G\Delta\{c\} = F\Delta\{u_1,\dots,u_{2s},u_n\}$ feasible, proceeding with the induction for u_n. If $G\Delta\{a,c\}$ is feasible, we have $G\Delta\{a,c\} = F\Delta\{u_1,\dots,u_{2s},u_{2s+1},u_n\}$ feasible, which inductively will give an edge $\{u_{2l+1},u_{2k+1}\}$ as above with $2k+1=n$ and $2l+1 \leq 2s+1$, and thus a blossom. □

We describe next an algorithm for finding an augmenting path or a blossom. Let u be such that $\{u,\overline{u}\} \notin \mathcal{L}_F$. Start a breadth first search at u that assigns levels to elements of E as follows. The element u is at level 1. If u_{2j-1} is at level $2j-1$, then put at level $2j$ all elements u_{2j} not at levels up to $2j-1$ such that $\{u_{2j-1},u_{2j}\}$ is an edge. If u_{2j} is at level $2j$, then put its mate $u_{2j+1} = \overline{u_{2j}}$ at level $2j+1$ if it is not at a level up to $2j$ and $\{u_{2j},u_{2j+1}\} \in \mathcal{L}_F$. The mate $\overline{u}$ of $u=u_1$ is omitted from the depth first search.

The algorithm terminates in one of four situations: (1) two distinct elements u_{2j},v_{2j} are mates, in which case a blossom has been found; (2) two distinct elements u_{2j+1},v_{2j+1} have an edge $\{u_{2j+1},v_{2j+1}\}$, in which case a blossom has also been found; (3) an element u_{2j} at level $2j$ is such that $\{u_{2j},\overline{u_{2j}}\} \notin \mathcal{L}_F$, in which case an augmenting path has been found; (4) an element u_{2j+1} at level $2j+1$ is a special element, in which case an augmenting path has been found.

Theorem 4.2 *The claims about having found an augmenting path or a blossom by the breadth first search in the four cases are correct.*

Proof. In case (1) the two paths from u_1 to u_{2j} and v_{2j} plus the mates u_{2j} and v_{2j} complete a blossom. In case (2) the two paths from u_1 to u_{2j+1} and v_{2j+1} plus the edge $\{u_{2j+1},v_{2j+1}\}$ complete a blossom.

In case (3) we have a path $u_1,\dots,u_{2j}$. We claim that it is an augmenting path, inductively on j. Let $\mathcal{K}$ be the subset of $\mathcal{L}$ consisting of the pairs $\{u_{2i},u_{2i+1}\}$ for each $2 \leq 2i \leq 2j-4$, and obtain M', $\mathcal{L}'$ by contracting $\mathcal{K}$. Let F' be the feasible set in M' corresponding to F in M. We apply Lemma 3.4 to F' with $a=u_1$, $b=u_{2j-2}$, $c=u_{2j-1}$, $d=u_{2j}$. We have that $F'\Delta\{a,b\}$ is feasible for M' since $F_1 = F\Delta\{u_1,u_2,\dots,u_{2j-3},u_{2j-2}\}$ is feasible inductively for M and $\mathcal{K}_{F_1} = \mathcal{K}$, by removing $\{u_{2j-2},u_{2j-1}\}$ from $\mathcal{L}$ and adding new mates for u_{2j-2},u_{2j-1}, so that $u_1,\dots,u_{2j-2}$ will be an augmenting path by induction. Also $F'\Delta\{c,d\}$ is feasible for M' since $F_2 = F\Delta\{u_{2j-1},u_{2j}\}$ is feasible for M by the definition of an edge, and $\mathcal{K}_{F_2} = \mathcal{K}$. If $F'\Delta\{a\}$ is feasible for M', then $F\Delta S_1$ is feasible for M with $u_1 \in S_1$ and $S_1 \subseteq \{u_1,u_2,\dots,u_{2j-3}\}$, and $\mathcal{K}_{F\Delta S_1} = \mathcal{K}$, so there must be an augmenting path contained in S_1. This subset however does not have all the edges and special vertices needed to satisfy the conditions in Theorem 4.1 when there exists an augmenting path as otherwise they would have been found in the breadth first search. If $F'\Delta\{a,c\}$ is feasible for M', then $F\Delta S_2$ is feasible for M with $u_1,u_{2j-1} \in S_2$ and $S_2 \subseteq \{u_1,u_2,\dots,u_{2j-3},u_{2j-1}\}$, and $\mathcal{K}_{F\Delta S_2} = \mathcal{K}$. In this case u_{2j-1} would have been reached earlier in the breadth first search. Similarly, if $F'\Delta\{a,d\}$ is feasible for M', then $F\Delta S_3$ is feasible for M with $u_1,u_{2j} \in S_3$ and $S_3 \subseteq \{u_1,u_2,\dots,u_{2j-3},u_{2j}\}$, and $\mathcal{K}_{F\Delta S_3} = \mathcal{K}$, so u_{2j} would have been reached earlier in the breadth first search. Therefore $F'\Delta\{a,b,c,d\}$ is feasible for M', and so $F\Delta S_4$ is feasible for M

with $u_1, u_{2j-2}, u_{2j-1}, u_{2j} \in S_4$ and $S_4 \subseteq \{u_1, \ldots, u_{2j}\}$, and $\mathcal{K}_{F\Delta S_4} = \mathcal{K}$. Furthermore S_4 must be equal to this subset, otherwise u_{2j} would have been reached earlier in the breadth first search. This proves we have obtained an augmenting path that replaces F with $F\Delta\{u_1, \ldots, u_{2j}\}$.

The proof in case (4) is analogous. We have a path $u_1, \ldots, u_{2j+1}$. We show again that it has the elements of some augmenting path. Let $\mathcal{K}$ be the subset of $\mathcal{L}$ consisting of the pairs $\{u_{2i}, u_{2i+1}\}$ for each $2 \leq 2i \leq 2j-2$, and obtain M', $\mathcal{L}'$ by contracting $\mathcal{K}$. Let F' be the feasible set in M' corresponding to F in M. We apply Lemma 3.2 to F' with $a = u_1$, $b = u_{2j}$, $c = u_{2j+1}$. We have that $F'\Delta\{c\}$ is feasible for M' since $F_3 = F\Delta\{u_{2j+1}\}$ is feasible for M because u_{2j+1} is a special element and $\mathcal{K}_{F_3} = \mathcal{K}$. Also $F'\Delta\{a, b\}$ is feasible for M' since $F_4 = F\Delta\{u_1, u_2, \ldots, u_{2j-1}, u_{2j}\}$ is feasible by the preceding case of an even length path. If $F'\Delta\{a\}$ is feasible for M', then $F\Delta S_5$ is feasible for M with $u_1 \in S_5$ and $S_5 \subseteq \{u_1, \ldots, u_{2j-1}\}$, and $\mathcal{K}_{F\Delta S_5} = \mathcal{K}$, so there must be an augmenting path contained in S_5, which does not have the edges in special vertices to satisfy the conditions in Theorem 4.1. If $F'\Delta\{a, c\}$ is feasible for M', then $F\Delta S_6$ is feasible for M with $u_1, u_{2j+1} \in S_6$ and $S_6 \subseteq \{u_1, \ldots, u_{2j-1}, u_{2j+1}\}$, and $\mathcal{K}_{F\Delta S_6} = \mathcal{K}$, which is not possible since u_{2j+1} would then have been reached earlier by the breadth first search. Therefore $F'\Delta\{a, b, c\}$ is feasible for M', and so $F\Delta S_7$ is feasible for M with $u_1, u_{2j}, u_{2j+1} \in S_7$ and $S_7 \subseteq \{u_1, \ldots, u_{2j+1}\}$, and $\mathcal{K}_{F\Delta S_7} = \mathcal{K}$. Furthermore S_7 must be equal to this subset, otherwise u_{2j} would have been reached earlier in the breadth first search. This proves we have obtained an augmenting path that replaces F with $F\Delta\{u_1, \ldots, u_{2j+1}\}$. □

5 Delta-Matroid Intersection without Equality

So far the argument has been carried in the full generality of arbitrary delta-matroids and the general parity problem. The arguments usually become more difficult with the introduction of blossoms, which can contain other blossoms, and this can lead to requiring the delta-matroid to have a presentation that is not only by means of an oracle, or some other special structure, such as in the case of linear or local delta-matroids [7, 3]. In special cases with restrictions involving the equal delta-matroid $M_=$ and the not-equal delta-matroid $M_{\neq}$, blossoms can be more easily handled. This leads to our main result.

Suppose the algorithm of Theorem 4.2 found a blossom as in Theorem 4.1. Restrict the breadth search for an augmenting path to the elements $w_1, \ldots, w_{2r}, \ldots, w_k$ of the blossom. When we restrict the breadth first search further by excluding some w_{2i}, w_{2i+1} we may find a smaller blossom or an augmenting path as in Theorem 4.2. We may thus assume this does not happen for the blossom under consideration.

Theorem 5.1 *There is a polynomial time algorithm using an oracle for any instance of delta-matroid parity on a delta-matroid M with pairing $\mathcal{L}$ such that no subset $\mathcal{K} \subset \mathcal{L}$ and pair $\{a, b\} \in \mathcal{L} \setminus \mathcal{K}$ are such that the delta-matroid M' obtained from M by contracting K has the not-equal delta-matroid $M_{\neq}$ on $\{a, b\}$ as a restriction.*

Proof. Let $w_1, \ldots, w_{2r}, \ldots, w_k$ be the blossom obtained above. Let $\mathcal{K}$ be the pairs $\{w_{2i}, w_{2i+1}\}$ for $2 \leq 2i \leq k-2$. Contracting $\mathcal{K}$ in M, we obtain M' with corresponding feasible set F'. Let $a = w_1$, $b = w_k$, $c = w_{k-1}$. The set $F'\Delta\{a, b\}$ is feasible by the augmenting path $w_1, \ldots, w_{2r-1}, w_k$ obtained after removing the pair $\{w_{k-1}, w_k\}$ from $\mathcal{L}$ and adding new mates for w_{k-1}, w_k, using Theorem 4.2. The set $F'\Delta\{a, c\}$ is feasible by the augmenting path $w_1, \ldots, w_{k-1}$ obtained also after removing the pair $\{w_{k-1}, w_k\}$ from $\mathcal{L}$ and adding new mates for w_{k-1}, w_k, using Theorem

4.2. Setting $G' = F'\Delta\{a\}$, we have that $G'\Delta\{b\}$ and $G'\Delta\{c\}$ are feasible, giving $M_{\neq}$ on $\{b,c\}$ as a restriction unless G' or $G'\Delta\{b,c\}$ is feasible.

If G' or $G'\Delta\{b,c\}$ is feasible, then there is an augmenting path involving a subset of $w_1,\ldots,w_k$, and this augmenting path cannot miss any $\{w_{2i}, w_{2i+1}\}$ by the choice of the blossom, so the elements $w_1,\ldots,w_k$ form an augmenting path in some order. The remaining case has $M_{\neq}$ on $\{b,c\}$ simulated by $G'\Delta\{b\}$ and $G'\Delta\{c\}$, contrary to assumption. □

We infer three results as special cases.

Lemma 5.2 *Let $M = (\{a,b,c,d\},\mathcal{F})$ be a delta-matroid that contracts to M' on $\{c,d\}$ using $\mathcal{K} = \{\{a,b\}\}$. If M' is either the $M_{=}$ or the $M_{\neq}$ delta-matroid, but M does not have M' on $\{c,d\}$ as a restriction, then $|F\Delta G|$ is even for all $F,G \in \mathcal{F}$.*

Proof. We have $M' = (\{c,d\},\{F, F\Delta\{c,d\}\})$ By assumption on M, we have a feasible F such that $F\Delta\{a,b,c,d\}$ is feasible, but $F\Delta\{a,b\}$, $F\Delta\{c,d\}$ are not feasible, and furthermore $F\Delta\{c\}$, $F\Delta\{d\}$, $F\Delta\{a,b,c\}$, $F\Delta\{a,b,d\}$ are not feasible. This guarantees that M' is not a restriction of M, and that contracting $\mathcal{K} = \{\{a,b\}\}$ gives M'.

If $F\Delta\{a\}$ is feasible, then taking $A = F\Delta\{a\}$, $B = F\Delta\{a,b,c,d\}$ and $x = b$ in the definition of delta-matroid yields a contradiction. If $F\Delta\{b\}$ is feasible, then taking $A = F\Delta\{b\}$, $B = F\Delta\{a,b,c,d\}$ and $x = a$ in the definition of delta-matroid yields a contradiction. If $F\Delta\{a,c,d\}$ is feasible, then taking $A = F\Delta\{a,c,d\}$, $B = F$ and $x = a$ in the definition of delta-matroid yields a contradiction. If $F\Delta\{b,c,d\}$ is feasible, then taking $A = F\Delta\{b,c,d\}$, $B = F$ and $x = b$ in the definition of delta-matroid yields a contradiction. Thus $|F\Delta G|$ is even for all feasible G. □

We shall make use of Wenzel's *strong exchange axiom* for even delta-matroids [12] $M = (E,\mathcal{F})$, which states that for all $A,B \in \mathcal{F}$ and $x \in A\Delta B$, there exists $y \in A\Delta B$ such that $A\Delta\{x,y\} \in \mathcal{F}$ and $B\Delta\{x,y\}$.

The first special case is as follows.

Theorem 5.3 *There is a polynomial time algorithm using an oracle for delta-matroid parity on delta-matroids M that do not have the not-equal delta-matroid $M_{\neq}$ as a restriction.*

Proof. It suffices that if M' is obtained from M by contracting $\mathcal{K}$, then M' does not have $M_{\neq}$ as a restriction either, so that the algorithm of Theorem 5.1 applies. Suppose $M_{\neq}$ on $\{c,d\}$ is obtained as a restriction after contracting $\mathcal{K} = \{\{a,b\}\}$, but not before. We may restrict M to $\{a,b,c,d\}$ and apply Lemma 5.2. We thus have a feasible $F = \{c\}$ such that $|F\Delta G|$ is even for all feasible G, and with $F\Delta\{a,b,c,d\}$ also feasible. By Wenzel's strong exchange axiom there are two other complementary feasible sets, say $F\Delta\{a,c\}$ and $F\Delta\{b,d\}$. Restricting M to feasible sets that do not contain either b or d, we obtain have two feasible sets $\{a\},\{c\}$ giving $M_{\neq}$ as a restriction on $\{a,c\}$ before contracting $\mathcal{K}$. □

The second special case is as follows.

Theorem 5.4 *There is a polynomial time algorithm using an oracle for delta-matroid parity on a delta-matroid M with pairing $\mathcal{L}$ for which there exist two disjoint sets of elements S,T each containing one element from each pair in $\mathcal{L}$ such that if the not-equal delta-matroid $M_{\neq}$ is a restriction of M on $\{a,b\}$, then both a and b are in S, and if the equal delta-matroid $M_{=}$ is a restriction of M on $\{a,b\}$, then at least one of a,b is in S.*

Proof. It suffices that if M' is obtained from M by contracting $\mathcal{K} \subseteq \mathcal{L}$, then M' also satisfies the property in the Theorem, so that M' does not have $M_{\neq}$ as a restriction on $\{a,b\}$ with at most one of a,b in S, and thus for $\{a,b\} \in \mathcal{L}$, and the algorithm of Theorem 5.1 applies. We show this by induction on $|\mathcal{K}|$.

Suppose $M_{=}$ or $M_{\neq}$ on $\{c,d\}$ is obtained as a restriction after contracting $\mathcal{K} = \{\{a,b\}\}$, with c and d in T. We may restrict M to $\{a,b,c,d\}$ and apply Lemma 5.2. We thus have a feasible F such that $|F\Delta G|$ is even for all feasible G, and with $F\Delta\{a,b,c,d\}$ also feasible. By Wenzel's strong exchange axiom there are two other complementary feasible sets, say $F\Delta\{a,c\}$ and $F\Delta\{b,d\}$. If a is in T, then F and $F\Delta\{a,c\}$ give $M_{=}$ or $M_{\neq}$ as a restriction on $\{a,c\}$ with both a and c in T, which is not possible by inductive hypothesis. Otherwise b is in T, and then F and $F\Delta\{b,d\}$ give $M_{=}$ or $M_{\neq}$ as a restriction on $\{b,d\}$ with both b and d in T, which is not possible by inductive hypothesis.

Suppose $M_{\neq}$ on $\{c,d\}$ is obtained as a restriction after contracting $\mathcal{K} = \{\{a,b\}\}$, with c in S and d in T, but not before. We may restrict M to $\{a,b,c,d\}$ and apply Lemma 5.2. We thus have a feasible $F = \{c\}$ or $F = \{d\}$ such that $|F\Delta G|$ is even for all feasible G, and with $F\Delta\{a,b,c,d\}$ also feasible. By Wenzel's strong exchange axiom there are two other complementary feasible sets, say $F\Delta\{a,c\}$ and $F\Delta\{b,d\}$. If $F = \{d\}$, then the two feasible sets $\{b\},\{d\}$ give a $M_{\neq}$ delta-matroid on $\{b,d\}$ with d in T, which is not possible by inductive hypothesis. If b is in T and $F = \{c\}$, then the two feasible sets F and $F\Delta\{b,d\}$ give an $M_{=}$ delta-matroid on $\{b,d\}$ with both b,d in T, which is not possible. Otherwise a is in T and $F = \{c\}$, and then the two feasible sets $\{a\},\{c\}$ give a $M_{\neq}$ delta-matroid on $\{a,c\}$ with a in T, which is not possible by inductive hypothesis. □

Corollary 5.5 *There is a polynomial time algorithm using an oracle for delta-matroid intersection on two delta-matroids M_1, M_2 where the delta-matroid M_1 is arbitrary, and the delta-matroid M_2 does not have either the equal delta-matroid $M_{=}$ or the not-equal delta-matroid $M_{\neq}$ as a restriction.*

Proof. Apply Theorem 5.4 with S consisting of the elements in M_1 and T consisting of the elements in M_2. □

The third special case is as follows.

Theorem 5.6 *There is a polynomial time algorithm using an oracle for delta-matroid parity on a delta-matroid M with pairing $\mathcal{L}$ for which there exist two disjoint sets of elements S, T each containing one element from each pair in $\mathcal{L}$ such that if the not-equal delta-matroid $M_{\neq}$ is a restriction of M on $\{a,b\}$, then either both a and b are in S or both a and b are in T, and if the equal delta-matroid $M_{=}$ is a restriction of M on $\{a,b\}$, then one of a,b is in S and the other one in T.*

Proof. It suffices that if M' is obtained from M by contracting $\mathcal{K} \subseteq \mathcal{L}$, then M' also satisfies the property in the Theorem, so that M' does not have $M_{\neq}$ as a restriction on $\{a,b\}$ with one of a,b in S and the other one in T, and thus for $\{a,b\} \in \mathcal{L}$, and the algorithm of Theorem 5.1 applies. We show this by induction on $|\mathcal{K}|$.

Suppose $M_{\neq}$ on $\{c,d\}$ is obtained as a restriction after contracting $\mathcal{K} = \{\{a,b\}\}$, with c in S and d in T, but not before. We may restrict M to $\{a,b,c,d\}$ and apply Lemma 5.2. We thus have a feasible $F = \{c\}$ such that $|F\Delta G|$ is even for all feasible G, and with $F\Delta\{a,b,c,d\}$ also feasible. By Wenzel's strong exchange axiome there are two other complementary feasible sets, say $F\Delta\{a,c\}$ and $F\Delta\{b,d\}$. If a is in T, then restricting M to feasible sets that do not contain either b or d, we obtain have two feasible sets $\{a\},\{c\}$ giving $M_{\neq}$ as a restriction on $\{a,c\}$ before contracting $\mathcal{K}$, which is not possible by inductive hypothesis. Otherwise b is in T, and restricting M to feasible sets that contain a and do not contain c, we obtain two feasible $\{a\}$ and $\{a,b,d\}$ giving $M_=$ as a restriction on $\{b,d\}$, which is not possible by inductive hypothesis.

Suppose $M_=$ on $\{c,d\}$ is obtained as a restriction after contracting $\mathcal{K} = \{\{a,b\}\}$, with c,d both in S, but not before. Say a is in S and b is in T. We may restrict M to $\{a,b,c,d\}$ and apply Lemma 5.2. We thus have a feasible $F = \emptyset$ or $F = \{a,b\}$ such that $|F\Delta G|$ is even for all feasible G, and with $F\Delta\{a,b,c,d\}$ also feasible. By Wenzel's strong exchange axiom there are two other complementary feasible sets, say $F\Delta\{a,c\}$ and $F\Delta\{b,d\}$. If $F = \emptyset$, then restricting M to feasible sets that do not contain either b or d, we obtain have two feasible sets $\emptyset, \{a,c\}$ giving $M_=$ as a restriction on $\{a,c\}$ with both a and c in S, which is not possible by inductive hypothesis. Otherwise $F = \{a,b\}$, and restricting M to feasible sets that contain a and do not contain c, we obtain two feasible $\{a,b\}$ and $\{a,d\}$ giving $M_{\neq}$ as a restriction on $\{b,d\}$ with b in T and d in S, which is not possible by inductive hypothesis. □

Corollary 5.7 *There is a polynomial time algorithm using an oracle for delta-matroid intersection on two delta-matroids M_1, M_2 that do not have the equal delta-matroid $M_=$ as a restriction. This generalizes matroid intersection, as matroids do not have the equal delta-matroid $M_=$ as a restriction.*

Proof. Apply Theorem 5.6 with S consisting of the elements in M_1 and T consisting of the elements in M_2. □

We note also that intersecting two matroids M_1 and M_2 is equivalent to intersecting M_1^- consisting of the independent sets of M_1, and M_2^+ consisting of the spanning sets of M_2. Furthermore both M_1^- and M_2^+ have neither $M_=$ nor $M_{\neq}$ as a restriction. Therefore all the results above generalize matroid intersection.

6 Bipartite Boolean Constraint Satisfaction

Corollary 5.7 also completes the classification of bipartite Boolean constraint satisfaction from [5, 11] as mentioned in the introduction. A *constraint* C on a set of Boolean variables X is a set of Boolean assignments x to the variables in X. A *restriction* of C by an assignment y to $Y \subseteq X$ is the constraint $C_{X,y}$ on the variables $X \setminus Y$ consisting of all assignments z such that if x is the assignment to X that agrees with y on Y and agrees with z on $X \setminus Y$, then x is in $C_{X,y}$. The *bipartite Boolean constraint satisfaction problem* on a set $\mathcal{C}$ of allowed constraints has an instance consisting of two sets of constraints S and T on subsets of a set of variables X, where each constraint in S or T corresponds to a constraint in $\mathcal{C}$ under some correspondence of variables, and each variable in X occurs in at most one constraint in S and at most one constraint in T. The aim is to assign Boolean values to the variables in X so as to satisfy the constraints in S and the constraints in T simultaneously. The bipartite case of Boolean constraint satisfaction differs from the general case with two occurrences per variable only when the equality constraint $\{00, 11\}$ is not an allowed constraint in $\mathcal{C}$. Let the inequality constraint be $\{10, 01\}$. A constraint is a delta-matroid if the collection of subsets of a set E with n elements defining a delta-matroid is viewed as a collection of assignments to n Boolean variables defining a constraint $\mathcal{C}$, where a 0 or 1 in a bit position corresponds to presence or absence of an element in the subset.

Theorem 6.1 *Every bipartite Boolean constraint satisfaction problem, with a set of allowed constraints closed under restriction, and where equality is not an allowed constraint, is one* *of Schaefer's polynomial cases, or polynomial by delta-matroid intersection without $M_=$ as a restriction, or is NP-complete. The polynomial cases remain polynomial even when the two sides of the bipartition are given by an oracle that answers whether a restriction $S_{X,y}$ or $T_{X,y}$ of either side of the bipartition is nonempty. This oracle result holds in the general case where equality is an allowed constraint for Schaefer's polynomial cases and for delta-matroids that do not contain inequality.*

Proof. The classification is obtained by Feder [5], and the remaining open case of delta-matroid intersection without $M_=$ as a restriction is polynomial by Corollary 5.7 using an oracle. When $M_=$ is allowed in delta-matroids, forbidding $M_{\neq}$ gives polynomiality by Theorem 5.3 using an oracle. The polynomial cases of Schaefer [11] are the following: (1) each constraint is a conjunction of 2-satisfiability clauses, (2) each constraint is a conjunction of Horn clauses, (3) each constraint is a conjunction of dual-Horn clauses, and (4) each constraint is a conjunction of linear equations modulo 2. An oracle in (1) allows us to obtain all the 2-satisfiability clauses and solve the problem. An oracle in (2) (resp. (3)) allows us to obtain all the variables forced to value 1 (resp. value 0) by some clause, and once no variable is forced, the remaining variables can be assigned value 0 (resp. value 1) if a solution exists.

For (4), we consider a candidate assignment x to the variables X and if this candidate assignment does not satisfy one of the two oracles, we obtain an assignment y to a subset of variables $Y \subseteq X$ such that y is a restriction of x and does not satisfy the oracle, yet every restriction of y to $|Y|-1$ variables in Y satisfies the oracle. This implies that the single linear equation involving precisely the variables in Y not satisfied by y must be satisfied by all assignments in the oracle. We then repeat the process for a candidate assignment x to the variables X satisfying this equation, and this assignment either satisfies both oracles or provides another equation. We proceed to add equations until a solution is found, or until the equations obtained so far are not satisfiable. Note that at most $n = |X|$ equations will be obtained, since each equation reduces by one the number of free variables, so the process terminates in polynomial time. □

We now proceed to the classification of bipartite Boolean constraint satisfaction problems in the case where the allowed constraint types may not be the same for both sides of the bipartition. Let $\mathcal{A}$ and $\mathcal{B}$ be two sets of constraint types. We say that $(\mathcal{A}, \mathcal{B})$ *simulates* constraint C on $\mathcal{A}$ if there exists an instance of bipartite Boolean constraint satisfaction with constraints from $\mathcal{A}$ in one side and constraints from $\mathcal{B}$ in the other side, with every variable constrained exactly once in the $\mathcal{A}$ side and constrained at most once in the $\mathcal{B}$ side, such that the variables that are not constrained in the $\mathcal{B}$ side are the variables of C, and the set of assignments of values to variables in C for which there exists a solution to this instance is the same as C. We say that $(\mathcal{A}, \mathcal{B})$ simulates constraint C on $\mathcal{B}$ if $(\mathcal{B}, \mathcal{A})$ simulates constraint C on $\mathcal{B}$.

Let $\mathcal{A}, \mathcal{B}, \mathcal{A}', \mathcal{B}'$ be sets of constraint types. We say that $(\mathcal{A}, \mathcal{B})$ *simulates* $(\mathcal{A}', \mathcal{B}')$ if $(\mathcal{A}, \mathcal{B})$ simulates every constraint $C \in \mathcal{A}'$ on $\mathcal{A}$ and simulates every constraint $C \in \mathcal{B}'$ on $\mathcal{B}$. We say that $(\mathcal{A}, \mathcal{B})$ is *closed under simulation* if whenever $(\mathcal{A}, \mathcal{B})$ simulates $(\mathcal{A}', \mathcal{B}')$ we have $\mathcal{A}' \subseteq \mathcal{A}$ and $\mathcal{B}' \subseteq \mathcal{B}$.

Theorem 6.2 *Let $\mathcal{A}, \mathcal{B}$ be sets of constraint types such that $(\mathcal{A}, \mathcal{B})$ is closed under simulation and both $\mathcal{A}, \mathcal{B}$ contain the single variable constraints $\{0\}$, $\{1\}$, and $\{0,1\}$. Then the bipartite Boolean constraint satisfaction with constraints from $\mathcal{A}$ in one side and from $\mathcal{B}$ in the other has (1) polynomial cases derived from Schaefer's classification; (2) polynomial cases derived from delta-matroid intersection in the case where neither side has equality and in the case where one side has neither equality nor inequality; (3) a polynomial case that combines 2-satisfiability and delta-matroid intersection. If a problem is not in cases (1), (2), or (3), then either (4) $\mathcal{A}$ is the same as $\mathcal{B}$ and* *consists of delta-matroids including equality, so the problem is a delta-matroid parity problem, or (5) the problem is NP-complete. The polynomial cases (1),(2),(3) remain polynomial in the oracle model as in Theorem 6.1.*

We define some specific constraint types on variables x, y, z. Let $[x = y]$ be $\{00, 11\}$. Let $[x \neq y]$ be $\{10, 01\}$. Let $[x \leq y]$ be $\{00, 01, 11\}$. Let $[x \vee y]$ be $\{10, 01, 11\}$. Let $[x = y = z]$ be $\{000, 111\}$. Let [1-3 x, y, z] be $\{100, 010, 001\}$. Let $[x \vee y \vee z]$ be $\{100, 010, 001, 110, 101, 011, 111\}$. Let $[x \leq y, z]$ be $\{000, 001, 010, 011, 111\}$. Let $[x + y + z = 0]$ be $\{000, 110, 101, 011\}$ and let $[x + y + z = 1]$ be $\{100, 010, 001, 111\}$. Let $[\approx x \vee y \vee z]$ be any constraint satisfying [1-3 x, y, z] $\subseteq [\approx x \vee y \vee z] \subseteq [x \vee y \vee z]$. Let $[\approx x \leq y, z]$ be any constraint satisfying $[x = y = z] \subseteq [\approx x \leq y, z] \subseteq [x \leq y, z]$. For these constraint types, we denote by $\overline{x}$ the complement of variable x, and by $\tilde{x}$ a literal that may be either x or $\overline{x}$. Feder [5] showed the following.

Lemma 6.3 *For a given constraint C, we have that $(\{C\}, \{\{0\}, \{1\}, \{0, 1\}\})$ simulates (1) $[x \vee y]$ or $[x \neq y]$ if C is not Horn; (2) $[\overline{x} \vee \overline{y}]$ or $[x \neq y]$ if C is not dual-Horn; (3) $[x \leq y]$ or $[x \vee y]$ or $[\overline{x} \vee \overline{y}]$ if C is not linear; (4) some $[\approx \tilde{x} \vee \tilde{y} \vee \tilde{z}]$ if C is not 2-SAT; and (5) some $[\approx \tilde{x} \leq \tilde{y}, \tilde{z}]$ if C is not a delta-matroid.*

If $X = x_1 x_2 \cdots x_k$ and $Y = y_1 y_2 \cdots y_k$ are k-bit vectors, write $X \leq Y$ if $x_i \leq y_i$ for all $1 \leq i \leq k$, write $X < Y$ if $X \leq Y$ and $X \neq Y$, and let $d(X, Y)$ be the Hamming distance between X and Y, that is, the number of bits $1 \leq i \leq k$ such that $x_i \neq y_i$.

Lemma 6.4 *For a given constraint C, (1) if $(\{C\}, \{\{0\}, \{1\}, \{0, 1\}\})$ simulates neither $[x = y]$ nor $[x \leq y]$, then for every $X, Y \in C$ with $X \leq Y$ we have that every Z such that $X \leq Z \leq Y$ satisfies $Z \in C$; (2) if $(\{C\}, \{\{0\}, \{1\}, \{0, 1\}\})$ simulates neither $[x \neq y]$ nor $[\overline{x} \vee \overline{y}]$, then there exists $X \in C$ such that $Z \leq X$ for every $Z \in C$; (3) if $(\{C\}, \{\{0\}, \{1\}, \{0, 1\}\})$ simulates neither $[x \neq y]$ nor $[x \vee y]$, then there exists $X \in C$ such that $X \leq Z$ for every $Z \in C$.*

Proof. For (1), if $X \leq X' < Y' \leq Y$ with $X', Y' \in C$ and $d(X', Y') \geq 2$, then there exists $X' < Z' < Y'$ such that $Z' \in C$. Otherwise we can consider the restriction C' of C to bit vectors T such that $t_i = b_i$ if $x'_i = y'_i = b_i$, and select i, j such that $x'_i < y'_i$ and $x'_j < y'_j$, so that the condition defined by C' on bit positions i, j is $[x_i = x_j]$. Thus by induction there exist $X = X^0 < X^1 < \cdots < X^k = Y$ with $d(X^i, X^{i+1}) = 1$ and each $X^i \in C$. Consider the restriction C' of C to bit vectors T such that $t_i = b_i$ if $x_i = y_i = b_i$, and say X^i has $x^i_j = 1$ for $1 \leq j \leq i$ and $x^i_j = 0$ for $i < j \leq k$. Assume inductively that if $X \leq T \leq X^i$ then $T \in C$. Suppose Z is such that $X \leq Z \leq X^{i+1}$ and $Z \notin C$ with $d(Z, X^{i+1})$ minimum. Then $z_{i+1} = 1$, and choosing $1 \leq j \leq i$ such that $z_j = 0$, we have that the bit vectors T obtained from Z by changing z_j or z_{i+1} or both are in C, thus giving $[x_{i+1} \leq x_j]$, completing the induction and the proof of (1).

For (2), if the condition does not hold, then there exist $X, Y \in C$ such that there is no $T \in C$ with $X < T$ or $Y < T$, and $d(X, Y) \geq 2$. Choose $X, Y \in C$ such that if we consider the restriction C' of C to bit vectors T with $t_i = b_i$ if $x_i = y_i = b_i$, then there is not $T \in C'$ with $X < T$ or $Y < T$, and $d(X, Y) \geq 2$ is minimum with this property. The minimality of $d(X, Y)$ implies that if $Z \in C'$ then $Z \leq X$ or $Z \leq Y$, otherwise some $Z' \in C'$ with $Z' \geq Z$ is such that there is no $T \in C'$ with $Z' < T$ and $Z' \neq X, Y$, so that $2 \leq d(X, Z') < d(X, Y)$, contrary to minimality. Let i, j be bit positions such that $x_i = 0, x_j = 1, y_i = 1, y_j = 0$. Then there is no $Z \in C'$ such that $z_i = z_j = 1$, so the condition defined by C' on bit positions i and j is either $[x_i \neq x_j]$ or $[\overline{x_i} \vee \overline{x_j}]$, proving (2). The proof for (3) is the same as for (2). □

A constraint C is *upward closed* if for every $X \in C$, if $X \leq Z$ then $Z \in C$, and *downward closed* if for every $X \in C$, if $Z \leq X$ then $Z \in C$. The upward closure of a constraint C is the constraint

$up(C)$ consisting of the bit vectors Z such that there exists $X \in C$ with $X \leq Z$. The downward closure of a constraint C is the constraint $down(C)$ consisting of the bit vectors Z such that there exists $X \in C$ with $Z \leq X$.

Lemma 6.5 *Let $\mathcal{A}, \mathcal{B}$ be as in the statement of Theorem 6.2, and suppose every constraint in $\mathcal{A}$ can be decomposed into an upward closed constraint, and constraints $\{0\}$. Then the Boolean constraint satisfaction problem with constraints from $\mathcal{A}$ in one side and from $\mathcal{B}$ in the other is polynomial in cases (1) for every $C \in \mathcal{B}$, $\mathrm{down}(C)$ can be decomposed into $\{0\}$, $\{0,1\}$ constraints, or every $C \in \mathcal{A}$ can be decomposed into $\{0\}$, $\{1\}$, $\{0,1\}$ constraints; (2) the constraints in $\mathcal{A}$ are delta-matroids, and for every $C \in \mathcal{B}$, $\mathrm{down}(C)$ is a delta-matroid; (3) the constraints in $\mathcal{A}$ are 2-SAT, and for every $C \in \mathcal{B}$, $\mathrm{down(C)}$ is 2-SAT. If we are not in cases (1),(2),(3), then $\mathcal{A}$ is NP-complete.*

Proof. We show that the problem reduces to the problem where $\mathcal{B}$ is replaced by $down(\mathcal{B})$ consisting of the constraints $down(C)$ for $C \in \mathcal{B}$. Given an instance of the problem with $\mathcal{B}$, if a constraint C in the $\mathcal{B}$ side has a variable x constrained by $\{0\}$ in $\mathcal{A}$, restrict C to bit vectors satisfying $x = 0$. If the resulting instance has a solution, then it also has a solution with each C in the $\mathcal{B}$ side replaced with $down(C)$. If the resulting instance has a solution with each C in the $\mathcal{B}$ side replaced with $down(C)$, then we may replace the X chosen from some $down(C)$ with a $Y \in C$ such that $X \leq Y$. This gives a solution with C, since replacing X with $Y \geq X$ will still satisfy the constraints in the $\mathcal{A}$ side, because these are upward closed.

If $down(C)$ in $down(\mathcal{B})$ can always be decomposed into $\{0\}$ and $\{0,1\}$ constraints, then it suffices to test the corresponding restriction of the $\mathcal{A}$ side. The same argument holds if $C \in \mathcal{A}$ can be decomposed into $\{0\}, \{1\}, \{0,1\}$ constraints.

Suppose $\mathcal{A}$ is delta-matroid, and $C \in \mathcal{A}$ cannot be decomposed into $\{0\}$, $\{1\}$, $\{0,1\}$ constraints. Then by (3) of Lemma 6.4 we can simulate $[x \neq y]$ or $[x \vee y]$ in the $\mathcal{A}$ side, and in fact we can simulate $[x \vee y]$ since $[x \neq y]$ is not upward closed. Given a constraint $D \in \mathcal{B}$ use for every variable x_i in D a corresponding condition $[x_i \vee y_i]$ in $\mathcal{A}$, to simulate a constraint C in $\mathcal{A}$ with variables y_i. The constraint C is obtained from $down(D)$ by complementing all bits. Since $\mathcal{A}$ is delta-matroid and closed under simulation, it follows that C is delta-matroid and thus $down(D)$ is delta-matroid. Once both sides are delta-matroids, the fact that the constraints in $\mathcal{A}$ are upward closed implies that they do not have $[x = y]$ or $[x \neq y]$ as a restriction, and the intersection of a delta-matroid without equality or inequality with an arbitrary delta-matroid is polynomial by Corollary 5.5.

Suppose $\mathcal{A}$ is 2-SAT, and $C \in \mathcal{A}$ cannot be decomposed into $\{0\}, \{1\}, \{0,1\}$ constraints. Then as in the delta-matroid case we get $[x \vee y]$ in the $\mathcal{A}$ side, and so for every constraint $D \in \mathcal{B}$ we get the constraint C obtained by complementing all bits of $down(B)$ in $\mathcal{A}$, so since $\mathcal{A}$ is 2-SAT and closed under simulation, it follows that $down(D)$ is 2-SAT as well. The problem is thus reduced to 2-SAT and therefore polynomial.

In the remaining case, $\mathcal{A}$ is not delta-matroid or 2-SAT, and for some C in $\mathcal{B}$ we have that $down(C)$ cannot be decomposed into $\{0\}, \{0,1\}$ constraints. Then by (3) of Lemma 6.4 we can simulate $[x \neq y]$ or $[\overline{x} \vee \overline{y}]$ in the $\mathcal{B}$ side, obtaining $[\overline{x} \vee \overline{y}]$ as the downward closure. By (4) of Lemma 6.3, we can simulate some $[\approx \tilde{x} \vee \tilde{y} \vee \tilde{z}]$ in the $\mathcal{A}$ side, and the only such constraint that is upward closed as required for the $\mathcal{A}$ side is $[x \vee y \vee z]$. By (5) of Lemma 6.3, we can simulate some $[\approx \tilde{x} \leq \tilde{y}, \tilde{z}]$ in the $\mathcal{A}$ side, and the only such constraint that is upward closed as required for the $\mathcal{A}$ side is $[\overline{x} \leq y, z]$ which we denote also by $[x \vee y, z]$.

We thus have $[x \vee y \vee z]$, $[x \vee y, z]$ in the $\mathcal{A}$ side. Combining these with conditions $[\overline{x} \vee \overline{x'}]$, $[\overline{y} \vee \overline{y'}]$, $[\overline{z} \vee \overline{z'}]$ on the $\mathcal{B}$ side gives corresponding $[\overline{x} \vee \overline{y} \vee \overline{z}]$, $[\overline{x} \vee \overline{y}, \overline{z}]$ in the $\mathcal{B}$ side. We do a reduction from 3-SAT. A 3-SAT clause that has both positive and negative literals can be decomposed into a clause that has only positive and a clause that has only negative literals, so that the two give the original

3-SAT clause by resolution. We already have positive and negative 3-SAT clauses simulated. By combining $[x \vee y \vee z]$ in $\mathcal{A}$ with $[\overline{z} \vee \overline{z'}]$ in $\mathcal{B}$ and with $[z' \vee z_1, z_2]$ in $\mathcal{A}$, we obtain $[x \vee y \vee z_1, z_2]$ in $\mathcal{A}$. This creates the multiple copies of variable z in the 3-SAT clause needed to combine with corresponding copies of $\overline{z}$ in 3-SAT clauses for $\mathcal{B}$, which can be obtained analogously. We thus have multiple copies of variables in clauses $[x \vee y \vee z]$ in $\mathcal{A}$ and clauses $[\overline{x} \vee \overline{y} \vee \overline{z}]$ in $\mathcal{B}$ as needed to complete the reduction and get NP-completeness. □

Note that the same result holds if we exchange upward closed with downward closed. A constraint C is Horn if every nonempty restriction C' of C has a least element. Thus in the first part of case (1) in Lemma 6.5, both $\mathcal{A}$ and $\mathcal{B}$ are dual-Horn. In the second part of case (1) the constraints in $\mathcal{A}$ decompose into monadic relations.

Lemma 6.6 *Let $\mathcal{A}, \mathcal{B}$ be as in the statement of Theorem 6.2, and suppose some constraint in $\mathcal{A}$ cannot be decomposed into an upward closed constraint, and constraints $\{0\}$, and some constraint in $\mathcal{A}$ cannot be decomposed into a downward closed constraint, and constraints $\{1\}$. Then we have the polynomial cases where $\mathcal{A}$ and $\mathcal{B}$ are both Horn, both dual-Horn, both 2-SAT, both linear, or with at most one side having $[x = y]$ the polynomial case of delta-matroids. In the remaining cases, either both sides are delta-matroids with $[x = y]$ and $\mathcal{A} = \mathcal{B}$, corresponding to delta-matroid parity, or the $\mathcal{A}$ side is neither Horn, dual-Horn, 2-SAT, linear, or delta-matroid.*

Proof. If $\mathcal{A}$ is a delta-matroid and $\mathcal{B}$ is not a delta-matroid, then by (5) of Lemma 6.3 we have some $[\approx \tilde{x} \leq \tilde{y}, \tilde{z}]$ in $\mathcal{B}$. In the cases where $\mathcal{A}$ has one of $[t = t']$, $[t \neq t']$, $[t \leq t']$, or both $[t \vee t']$ and $[\overline{t} \vee \overline{t'}]$, then combining such conditions for t or t' being x, y, z and corresponding x', y', z' we get $[\approx \tilde{x} \leq \tilde{y}, \tilde{z}]$ in $\mathcal{A}$, which is not a delta-matroid, contrary to assumption. We thus have of these choices for t, t' either just $[t \vee t']$ or just $[\overline{t} \vee \overline{t'}]$. By cases (1),(2),(3) of Lemma 6.4, we have that $\mathcal{A}$ decomposes into constraints $\{0\}$, $\{1\}$, and just upward closed constraints or just downward closed constraints, contrary to assumption.

If both $\mathcal{A}$ and $\mathcal{B}$ are delta-matroids, and $\mathcal{A}$ does not have $[x = y]$, then either $\mathcal{B}$ does not have $[x = y]$ either and the problem is polynomial by Corollary 5.7; or $\mathcal{B}$ does have $[x = y]$, in which case $\mathcal{A}$ does not have $[x \neq x']$ since using also $[y \neq y']$ would give $[x' = y']$ in $\mathcal{A}$ as well, so the problem is polynomial by Corollary 5.5. If both sides have $[x = y]$, then every constraint in $\mathcal{A}$ can also be obtained in $\mathcal{B}$ and viceversa, so $\mathcal{A} = \mathcal{B}$ and we have a class of delta-matroid parity problems.

If $\mathcal{A}$ is 2-SAT and $\mathcal{B}$ is not 2-SAT, then by (4) of Lemma 6.3 we have some $[\approx \tilde{x} \vee \tilde{y} \vee \tilde{z}]$ in $\mathcal{B}$. In the cases where $\mathcal{A}$ has one of $[t = t']$, $[t \neq t']$, $[t \leq t']$, or both $[t \vee t']$ and $[\overline{t} \vee \overline{t'}]$, then combining such conditions for t or t' being x, y, z and corresponding x', y', z' we get $[\approx \tilde{x} \vee \tilde{y} \vee \tilde{z}]$ in $\mathcal{A}$, which is not 2-SAT, contrary to assumption. We thus have of these choices for t, t' either just $[t \vee t']$ or just $[\overline{t} \vee \overline{t'}]$. By cases (1),(2),(3) of Lemma 6.4, we have that $\mathcal{A}$ decomposes into constraints $\{0\}$, $\{1\}$, and just upward closed constraints or just downward closed constraints, contrary to assumption. If both sides are 2-SAT the problem is polynomial.

If $\mathcal{A}$ is linear and $\mathcal{B}$ is not linear, and $\mathcal{A}$ is not 2-SAT, then we have in $\mathcal{A}$ a linear constraint $[x+y+z = 0]$ or $[x+y+z = 1]$, and in $\mathcal{B}$ by (3) of Lemma 6.3 we have $[z \leq z']$, or $[z \vee z']$, or $[\overline{z} \vee \overline{z'}]$. combining these with $[x' + y' + z' = 0]$ or $[x' + y' + z' = 1]$ in $\mathcal{A}$ gives a constraint on x, y, x', y' that is not linear, contrary to assumption. If both sides are linear the problem is polynomial.

If $\mathcal{A}$ is Horn and $\mathcal{B}$ is not Horn, then in $\mathcal{B}$ we get $[x \neq y]$ or $[x \vee y]$ by (1) of Lemma 6.3. Since $\mathcal{A}$ does not decompose into $\{1\}$ constraints and downward closed constraints by assumption, we have by (1),(3) of Lemma 6.4 that $\mathcal{A}$ has $[x = y]$, or $[x \leq y]$, or $[x \neq y]$, or $[x \vee y]$, yet the last two are not Horn, so $\mathcal{A}$ must have $[x = y]$ or $[x \leq y]$. Combining these with $[x \neq y]$ or $[x \vee y]$ in $\mathcal{B}$ gives $[x' \vee y']$ in $\mathcal{A}$, which is not Horn, contrary to assumption. If both sides are Horn the problem is polynomial. The case of dual-Horn is identical. □

We now consider situations where the last case in the preceding lemma gives NP-completeness.

Lemma 6.7 *Let $\mathcal{A}, \mathcal{B}$ be as in the statement of Theorem 6.2. Assume $\mathcal{A}$ is not Horn, dual-Horn, 2-SAT, linear, or delta-matroid. Assume also $\mathcal{B}$ contains either $[x = y]$ or $[x \leq y]$. Then the bipartite constraint satisfaction problem is NP-complete.*

Proof. If $\mathcal{B}$ contains $[x = y]$, then either we have $[x \neq y]$ in $\mathcal{A}$ which combines with a condition from (3) of Lemma 6.3 to give $[x \leq y]$ in $\mathcal{B}$, or we have both $[x \vee y]$ and $[\overline{x} \vee \overline{y}]$ by (1),(2) of Lemma 6.3 to also give $[x \leq y]$ in $\mathcal{B}$.

If $\mathcal{B}$ contains $[t \leq t']$, combining this with $[\approx \tilde{x} \vee \tilde{y} \vee \tilde{z}]$ from $\mathcal{A}$ by (4) of Lemma 6.3 gives $[\tilde{x} \vee \tilde{y} \vee \tilde{z}]$ in $\mathcal{B}$. Combining with $[\approx \tilde{x} \leq \tilde{y}, \tilde{z}]$ from $\mathcal{A}$ by (5) of Lemma 6.3 gives $[\tilde{x} \leq \tilde{y}, \tilde{z}]$ in $\mathcal{B}$. Since $\mathcal{A}$ contains either $[t \neq t']$ or both $[t \vee t']$ and $[\overline{t} \vee \overline{t'}]$ from (1) and (2) of Lemma 6.3, we also get $[\tilde{x} \vee \tilde{y} \vee \tilde{z}]$ and $[\tilde{x} \leq \tilde{y}, \tilde{z}]$ in $\mathcal{A}$. Using $[x \leq y]$ from $\mathcal{B}$ and $[t \neq t']$ or both $[t \vee t']$, $[\overline{t} \vee \overline{t'}]$ from $\mathcal{A}$ we get both $[x \vee y]$ and $[\overline{x} \vee \overline{y}]$ in $\mathcal{B}$, that is all three kinds of 2-SAT clauses in $\mathcal{B}$. We can thus replace each $\tilde{x}$ in the conditions of $\mathcal{A}$ with any choice out of x or $\overline{x}$ using these clauses in $\mathcal{B}$. Since $\mathcal{A}$ contains $[\tilde{x} \vee \tilde{y} \vee \tilde{z}]$, we have that $\mathcal{A}$ contains $[x \vee y]$ or $[\overline{x} \vee \overline{y}]$. Say $\mathcal{A}$ contains $[\overline{x} \vee \overline{y}]$, and then using $[x \vee y \vee z]$ and $[x \vee y, z]$ in $\mathcal{B}$ gives NP-completeness as in the last part of the proof of Lemma 6.5. □

In the remaining case, neither $\mathcal{A}$ nor $\mathcal{B}$ contains either $[x = y]$ or $[x \leq y]$. By (1) of Lemma 6.4, this implies that every constraint C in $\mathcal{A}$ or $\mathcal{B}$ satisfies $C = up(C) \cap down(C)$. Furthermore $\mathcal{A}$ is not delta-matroid, so it contains a constraint $[\approx \tilde{x} \leq \tilde{y}, \tilde{z}]$, and by the property just stated this must be either $[x \vee y, z]$, $[x \vee y, z] \setminus \{111\}$, $[\overline{x} \vee \overline{y}, \overline{z}]$, $[\overline{x} \vee \overline{y}, \overline{z}] \setminus \{000\}$. Say by symmetry it is either $[x \vee y, z]$ or $[x \vee y, z] \setminus \{111\}$. Then $\mathcal{A}$ contains $[x \vee y]$. Since $\mathcal{B}$ is not dual-Horn, it contains either $[\overline{x} \vee \overline{y}]$ or $[x \neq y]$, and in this last case it contains $[\overline{x} \vee \overline{y}]$ as well by combination with $[x \vee y]$ from $\mathcal{A}$. Combining $[\overline{x} \vee \overline{y}]$ from $\mathcal{B}$ with either $[x \vee y, z]$ or $[x \vee y, z] \setminus \{111\}$ from $\mathcal{A}$, we get $[\overline{x} \vee \overline{y}, \overline{z}]$ in $\mathcal{B}$, and thus $[x \vee y, z]$ in $\mathcal{A}$. Furthermore, we get the complement of $down(D)$ in $\mathcal{A}$ for every constraint D in $\mathcal{A}$, so by Lemma 6.5 the problem is NP-complete unless $down(D)$ is 2-SAT. Similarly, we get the complement of $up(C)$ in $\mathcal{B}$ for every constraint C in $\mathcal{A}$, and by Lemma 6.5 the problem is NP-complete unless $up(C)$ is 2-SAT.

If $up(D)$ is not delta-matroid, then by (5) of Lemma 6.3 it contains $[x \vee y, z]$ so D contains $[x \vee y, z]$ or $[x \vee y, z] \setminus \{111\}$. Then by the preceding argument exchanging $\mathcal{A}$ and $\mathcal{B}$, we have that the problem is NP-complete unless $up(D)$ and $down(C)$ are 2-SAT. In this last case, since $C = up(C) \cap down(C)$ and $D = up(D) \cap down(D)$, the whole problem is 2-SAT.

Thus in the remaining case $up(D)$ is a delta-matroid, and by the same argument $down(C)$ is a delta-matroid, while $up(C)$ and $down(D)$ are 2-SAT, for every C in $\mathcal{A}$ and D in $\mathcal{B}$. By complementing $\mathcal{A}$, this problem is more easily viewed as having an instance consisting of M such that $up(M)$ is a delta-matroid and $down(M)$ is 2-SAT, with the variables partitioned into pairs x, y that must satisfy $[x \neq y]$ in a solution.

Eliminate any variable x such that M has no bit-vector with $x = 1$ (resp $x = 0$), while setting $y = 1$ (resp $y = 0$) for the corresponding variable in the pair $[x \neq y]$. We may solve the $down(M)$ 2-SAT part with constraints $[x \neq y]$ and obtain a solution X if one exists. We may also solve the $up(M)$ delta-matroid part with constraints $[x \neq y]$ and obtain a solution Y if one exists, by Corollary 5.7. If X is in $up(M)$, then X is in $M = up(M) \cap down(M)$ and we are done. If X is not in $up(M)$, then there exists a $T \geq X$ with T not in $up(M)$ such that every $U > T$ is in $up(M)$. Let S be the set of variables x that have value 0 in T, called a *flat* of $up(M)$. There is no element of $up(M)$ that has all variables x in S with value 0. For every x in S, there is a least element of M that has $x = 1$ and with $y = 0$ for all other y in S.

We claim that X' obtained from X by changing $x = 0$ to $x = 1$ is also in $down(M)$. Otherwise X' fails to satisfy some 2-SAT clause involving x, say $[\overline{x} \vee \overline{z}]$ for some z not in S. Then an element V of M with $x = 1$ and $y = 0$ for all other y in S also has $z = 0$. Restricting M to the variables in $S \cup \{z\}$, to obtain M', we have that $up(M')$ is a delta-matroid, and contains V with a single variable $x = 1$. Let W be a least vector in M' having $z = 1$, and let y be some other variable that has $y = 1$ in W. Restrict M' by setting all variables other than x that have value 0 in W to value 0, thus obtaining M''. We have in M'' vectors with $yz = 00$ and vectors with $yz = 11$, but no vector with $yz = 01$ in M'', giving either $[z \leq y]$ or $[z = y]$, contrary to assumption. This proves the claim.

Note that the solution Y obtained above must have some x in S with value $x = 1$. Thus we may just obtain X' by trying all choices of variables x that have $x = 0$ in X and $x = 1$ in Y, until X' obtained by changing x does not fail to satisfy the 2-SAT clauses in $down(M)$. We may then change the mate that was linked to x by $[x \neq y]$ from $y = 1$ to $y = 0$ to obtain a new solution X'' in $down(M)$ which is closer to Y. Repeating the process, we eventually reach some X'' in $down(M)$ that is also in $up(M)$, since otherwise we keep getting closer to Y, and Y is in $up(M)$. We thus obtain from X and Y some X'' such that $X'' \in down(M) \cap up(M) = M$ and satisfies all conditions $[x \neq y]$ as well.

This algorithm completes the proof of Theorem 6.2.

Theorem 6.8 *Let M be an upward closed delta-matroid, let R be a collection of 2-SAT clauses $[\overline{x}]$, $[\overline{x} \vee \overline{y}]$ such that no flat of M with at least two elements meets a clause $[\overline{x} \vee \overline{y}]$ in exactly one element. Then one can solve the constraints given by M, R, and a pairing with conditions $[x \neq y]$ in polynomial time.*

We now obtain a full classification for k-partite Boolean constraint satisfaction for $k \geq 3$.

Theorem 6.9 *Let $\mathcal{A}_1, \ldots, \mathcal{A}_k$ be sets of constraint types each containing the single variable constraints $\{0\}$, $\{1\}$, $\{0,1\}$ and at least one constraint that does not decompose into these, with $k \geq 3$. Then the k-partite Boolean constraint satisfaction problem with constraints from $\mathcal{A}_i$ in part i and each variable participating in only one constraint from each part i is either polynomial time solvable using an oracle or NP-complete.*

Proof. If some $\mathcal{A}_i$ contains a constraint that is not a delta-matroid, say $\mathcal{A}_1$, then the problem defined by $\mathcal{A}_1$ and $\mathcal{A}_2$ is either polynomial time solvable using an oracle or NP-complete by Theorem 6.2. The NP-completeness of this subproblem implies the NP-completeness of the entire problem, while if the subproblem is polynomial then the algorithm simulates an oracle for the solutions of the subproblem, thus giving a new problem where the parts $\mathcal{A}_1$ and $\mathcal{A}_2$ have been combined into a single part $\mathcal{A}'$ that contains a constraint that is not a delta-matroid, thus reducing the analysis for k to $k-1$.

In the remaining case all $\mathcal{A}_i$ are delta-matroids. If at most one $\mathcal{A}_i$ contains $[x = y]$, say $\mathcal{A}_1$, then all $\mathcal{A}_j$ for $j > 1$ not containing $[x = y]$ also do not contain $[x \neq y]$ unless $\mathcal{A}_1$ does not contain $[x = y]$, otherwise $[x = y]$ could be simulated on $\mathcal{A}_j$ as well. We may thus intersect the delta-matroids from $\mathcal{A}_1$ and $\mathcal{A}_2$ by Corollaries 5.5 and 5.7, again giving a new problem where the parts $\mathcal{A}_1$ and $\mathcal{A}_2$ have been combined into a single part $\mathcal{A}'$ for which an oracle can be simulated, thus reducing the analysis for k to $k-1$.

If all $\mathcal{A}_i$ are delta-matroids, at least one $\mathcal{A}_i$ contains $[x = y]$, say $\mathcal{A}_1$, and at least one $\mathcal{A}_i$ does not contain $[x = y]$, say $\mathcal{A}_2$, then we may again combine $\mathcal{A}_1$ and $\mathcal{A}_2$ into a single part $\mathcal{A}'$ by Corollary 5.5, and this part is not a delta-matroid by a constraint on x, y, z given by $[x = y]$ in $\mathcal{A}_1$

and one of $[x \neq z]$, $[x \leq z]$, $[x \vee z]$, $[\overline{x} \vee \overline{z}]$ in $\mathcal{A}_2$, thus reducing the analysis to an earlier case from k to $k-1$.

Finally, if all $\mathcal{A}_i$ are delta-matroids and contain $[x = y]$, then combining $[x = y]$ in $\mathcal{A}_1$, $[y = z]$ in $\mathcal{A}_2$, and $[x = x'], [y = y'], [z = z']$ in $\mathcal{A}_3$ gives $[x' = y' = z']$ in $\mathcal{A}_3$, contrary to the assumption that $\mathcal{A}_3$ is a delta-matroid. □

7 Delta-Matroid Parity without Oracle

For the remaining open cases of the general problem where equality is an allowed constraint, namely cases where all constraints are given by delta-matroids, we note that not all known polynomial cases remain polynomial in the oracle model. The coindependent delta-matroid case from Feder [5] has an algorithm polynomial in $n2^k$, where n is the number of variables and k is the maximum number of variables per constraint, and has a lower bound exponential in k if a constraint on k variables is given by an oracle. There are other cases, such as local delta-matroids [3], that remain polynomial with an oracle.

We generalize the case of coindependent delta-matroids. A delta-matroid $M = (E, \mathcal{F})$ is a *zebra delta-matroid* if there exist integers $0 \leq r \leq s \leq |E|$ such that: (1) For all feasible sets $F \in \mathcal{F}$, $r \leq |F| \leq s$; (2) For all $A \subseteq E$ with $|A| \in \{r, s\}$, A is a feasible set, that is, $A \in \mathcal{F}$; and (3) For all $A \subseteq E$ with $r < |A| < s$, either $A \in \mathcal{F}$, or for all $B \subseteq E$, if $|A \Delta B| = 1$ then $B \in \mathcal{F}$. Zebra delta-matroids generalize the delta-matroids arising in the general factor problem.

A zebra delta-matroid is a *coindependent delta-matroid* if $r \in \{0, 1\}$ and $s \in \{|E| - 1, |E|\}$. Coindependent delta-matroids were studied in [5].

Theorem 7.1 *Delta-matroid parity on a coindependent delta-matroid with $|E| = 2k$ with oracle has a lower bound of 2^k on the number of queries to the oracle.*

Proof. Let $E = \{x_1, \ldots, x_{2k}\}$ and $\mathcal{L} = \{\{x_{2i-1}, x_{2i}\} : 1 \leq i \leq k\}$. Consider a set $A \subseteq E$ such that for all $1 \leq i \leq k$, $x_{2i-1} \in A$ if and only if $x_{2i} \in A$. Let $\mathcal{F}$ be the set of all subsets of E of odd cardinality plus A, which is of even cardinality. While an algorithm has queried fewer than 2^k sets $B \subseteq E$ such that for all $1 \leq i \leq k$, $x_{2i-1} \in B$ if and only if $x_{2i} \in B$, the oracle may answer that $B \notin \mathcal{F}$, and only when the 2^k-th such B is queried set $A = B$, giving the answer to the problem. □

We prove a counterpart to this lower bound.

Theorem 7.2 *Suppose $M = (E, \mathcal{F})$ with $|E| = n$ is the direct sum of zebra delta-matroids $M_i = (E_i, \mathcal{F}_i)$ with $|E_i| \leq k$, $|\mathcal{F}_i| \leq f$. Then delta-matroid parity on M, $\mathcal{L}$, can be solved in time $O(n^3 f)$.*

We successively simplify the problem.

Lemma 7.3 *The problem reduces to the case where we have a feasible F and only a single $\{a, b\} \in \mathcal{L}$ such that $\{a, b\} \notin \mathcal{L}_F$, and one of the M_i has $E_i = \{b\}$.*

Proof. The problem reduces to finding an augmenting path. For each choice of an element a with which to start the augmenting path given $F \in \mathcal{F}$, so that $\{a, b\} \in \mathcal{L}$ and $\{a, b\} \notin \mathcal{L}_F$, let S be the set of elements c such that $\{c, d\} \in \mathcal{L}$ and $\{c, d\} \notin \mathcal{L}_F$ for some element d. For each $c \in S$, let $M'_c = (\{\overline{c}\}, \mathcal{F}'_c)$, where $\mathcal{F}'_c = \{\emptyset, \{\overline{c}\}\}$ if $c \neq a, b$, $\{\overline{b}\} \in \mathcal{F}'_b$ if and only if $b \in F$, $\emptyset \in \mathcal{F}'_b$ if and and only if $b \notin F$, $\{\overline{a}\} \in \mathcal{F}'_a$ if and only if $a \notin F$, and $\emptyset \in \mathcal{F}'_a$ if and only if $a \in F$. Let M' be the direct sum of the M_i and the M'_c. Extend the feasible F for M to a feasible F' for M' by including $\overline{c} \in S$ in F' if and only if $c \in F$ for $c \neq a$, and including $\overline{a}$ in F' if and only if $a \notin F$. Let $\mathcal{L}'$ consist of the pairs $\{c,d\} \in \mathcal{L}$ such that $c, d \notin S$, and the pairs $\{c, \overline{c}\}$ for $c \in S$. Thus the only $\{c,d\} \in \mathcal{L}'$ such that $\{c,d\} \notin \mathcal{L}'_{F'}$ is $\{c,d\} = \{a, \overline{a}\}$, and the augmenting paths for $M, \mathcal{L}$ started at a correspond to the augmenting paths for $M', \mathcal{L}'$, which must start at a. □

Consider the problem in the form of Lemma 7.3, with zebra delta-matroids $M_i = (E_i, \mathcal{F}_i)$ having corresponding r_i, s_i. Let $M'_i = (E_i, \mathcal{F}'_i)$, where $\mathcal{F}'_i$ consists of the sets $F'_i \subseteq E_i$ such that $r_i \leq |F'_i| \leq s_i$ and $F'_i \Delta (F \cap E_i)$ is of even size.

Lemma 7.4 *The problem reduces to a problem where all but one of the M_i have been replaced by M'_i and the conditions of Lemma 7.3 are also met.*

Proof. The augmenting path starting at a must end in some $M_0 = (E_0, \mathcal{F}_0)$. Replace all $M_i \neq M_0$ with M'_i, to obtain M'. Since every $M_i \neq M_0$ has an even number of elements in the augmenting path starting at a, it follows that this augmenting path is also an augmenting path in M'. Conversely, suppose we have an augmenting path in M', starting at a. Let the augmenting path be $a = x_1, \ldots, x_{2t+1}$ with $x_{2t+1} \in E_0$. Either this is also an augmenting path for M, or there exists an $1 \leq j \leq t$ such that $F \Delta \{x_1, \ldots, x_{2j-2}\}$ is feasible for M but $F \Delta \{x_1, \ldots, x_{2j}\}$ is not feasible for M. In this last case, since each M_i is a zebra delta-matroid, we have that x_{2j-1} is in some M_i with r_i, s_i and $(F \Delta \{x_1, \ldots, x_{2j-1}\}) \cap E_i$ has size $r_i \leq u \leq s_i$, so the fact that $F \Delta \{x_1, \ldots, x_{2j}\}$ is not feasible implies that $F \Delta \{x_1, \ldots, x_{2j-1}\}$ is feasible, giving an augmenting path $a = x_1, \ldots, x_{2j-1}$ for M. □

Consider the problem in the form of Lemma 7.4.

Lemma 7.5 *The problem reduces to graph matching.*

Proof. So far, we have a single $M_0 = (E_0, \mathcal{F}_0)$ not of the form of the M'_i, with $|E_0| \leq k$. We may then consider each of the at most f feasible sets $F_0 \in \mathcal{F}_0$ such that $|(F \cap E_0) \Delta F_0|$ is odd, and replace M_0 with $M'_0 = (E_0, \{F_0\})$, which decomposes into $|E_0| \leq k$ zebra delta-matroids, giving at most k unmatched pairs plus the pair $\{a, b\}$.

Now the problem has delta-matroids $M'_i = (E_i, \mathcal{F}'_i)$. with $\mathcal{F}'_i$ consisting of all F'_i with $r \leq |F'_i| \leq s$ and both $t = s - r, |F'_i| - r$ even. Define a graph G consisting of a clique K on t vertices, an independent set I on r vertices, a complete bipartite graph with the $r + t$ vertices in $K \cup I$ in one side and some additional $r + t$ vertices forming a set U in the other side. To match all vertices in $K \cup I \cup U$, we must have $r \leq r + 2j \leq r + t$ vertices in U matched to vertices in $K \cup I$, corresponding to a choice of $r + 2j$ elements from E_i forming some $F'_i \in \mathcal{F}'_i$. We may then join the sets $U = U_i$ for each M'_i with edges corresponding to the pairs $\{a, b\} \in \mathcal{L}$.

We may assume a is not in M_0. We look for an F'-augmenting path in the resulting graph starting at a for $F' = F \Delta ((F \Delta F_0) \cap E_0)$. The augmenting path $a, \ldots, x_1$ will have $\overline{x_1} \in (F \cap E_0) \Delta F_0$, and either $(F \cap E_0) \Delta \{\overline{x_1}\}$ is feasible for M_0, in which case we replace F with $F' = F \Delta \{a, \ldots, x_1, \overline{x_1}\}$, or we replace F with $F'' = F' \Delta \{x_2\}$ for some $x_2 \in (F' \cap E_0) \Delta F_0$ such that $(F' \cap E_0) \Delta \{x_2\}$ is feasible for M_0, set $a = \overline{x_2}$ and proceed to look for an augmenting path starting at a. In the end, we will either have $F \cap E_0 = F_0$ or have found a shorter augmenting path by Lemma 7.4. □

The graph of Lemma 7.5 has $O(n)$ vertices and $m = O(nk)$ edges and requires finding at most f augmentations in a graph if we go through the F_0 with $d_0 = |(F \cap E_0) \Delta F_0|$ in order of increasing d_0. Each augmentation can be done in time $O(m)$, giving a total time $O(mf) = O(nkf)$ for the problem of Lemma 7.4. This complexity can be reduced by only implicity maintaining the graph corresponding to of each M_i, so that only the $O(|E_i|)$ times that M_i is visited are counted, and the search for an augmenting path takes $O(n)$ time. Each of the sets in $\mathcal{F}_0$ is considered at most k times while finding augmenting paths, once for each element in E_0 to be included. Thus the problem of Lemma 7.4 is solved in $O(nf)$ time. The problem of Lemma 7.3 can be solved in time $O(n^2 f)$ by considering the at most n possible choices of M_0. Testing the at most n unmatched pairs to find a maximum number of augmentations for the original problem can be done in time $O(n^3 f)$, solving the original problem and proving Theorem 7.2.

Of course, for the general factor problem, which can be viewed as consisting of zebra delta-matroids such that if A is feasible then every B with $|B| = |A|$ is also feasible, we can let M'_0 consist of the sets of a given size, and only two sizes need to be considered, namely the sizes p, q such that $p < v = |F \cap E_0| < q$ that give the least odd values for $v - p, q - v$. This costs of a factor of k for at most k augmentations instead of f, giving a bound of $O(n^3 k)$ on the running time. See also Cornuejols [2] for a more efficient algorithm for the general factor problem.

Istrate [8] defined compact delta-matroids by combining the delta-matroids of the general factor problem in a star arrangement. More generally, we can combine zebra delta-matroids in a tree configuration. Formally, define $M = (E, \mathcal{F})$ to be a *zebra-compact delta-matroid* inductively if there exists a zebra delta-matroid $M_1 = (E_1 \cup \{a\}, \mathcal{F}_1)$ and a zebra-compact delta-matroid $M_2 = (E_2 \cup \{b\}, \mathcal{F}_2)$ such that M is obtained from M_1, M_2 by linking a and b and contracting $K = \{\{a, b\}\}$; also the direct sum of a zebra delta-matroid and a zebra-compact delta-matroid is a zebra-compact delta-matroid, and every zebra delta-matroid is a zebra-compact delta-matroid.

Theorem 7.6 *A zebra-compact delta-matroid $M = (E, \mathcal{F})$ with $|E| = k$ can be recognized and decomposed into zebra delta-matroids in time $O(c^{k^2})$ for some constant c.*

Proof. We can test each of the 2^k possible decompositions $E_1 \subseteq E$, $E_2 = E \setminus E_1$, each in time $O(c^k)$ for some constant c. If there exist two elements $A \cup B, A' \cup B' \in \mathcal{F}$, where $A, A' \subseteq E_1$ and $B, B' \subseteq E_2$, such that $A \cup B', A' \cup B \notin \mathcal{F}$, then this determines uniquely the feasible sets of M_1, M_2, namely the possible choices of B'', B''' such that $A \cup B'', A' \cup B''' \in \mathcal{F}$ give feasible sets $B'' \cup \{b\}, B''' \in \mathcal{F}_2$ or $B'', B''' \cup \{b\} \in \mathcal{F}_2$, and similarly for $\mathcal{F}_1$. After verifying this decomposition, we proceed inductively.

Otherwise, unless M is a direct product of $M_1 = (E_1, \mathcal{F}_1)$ and $M_2 = (E_2, \mathcal{F}_\in$, we only have $A \cup B, A \cup B', A' \cup B' \in \mathcal{F}$ but $A' \cup B \notin \mathcal{F}$. Then we have $B \cup \{b\}, B' \in \mathcal{F}_2$, and possibly $B' \cup \{b\} \in \mathcal{F}_2$ (or equivalently $B, B' \cup \{b\} \in \mathcal{F}_2$ and possibly $B' \in \mathcal{F}_2$). All the possibly included subsets must be included either for $\mathcal{F}_1$ or for $\mathcal{F}_2$, say for $\mathcal{F}_1$, and it can then be shown that including them all or none for $\mathcal{F}_2$ will allow the decomposition to proceed if some subset of them allows the decomposition to proceed. However, we may then not get a zebra delta-matroid for each delta-matroid that is not further decomposed. It may at that point be decided whether to include the possibly included subsets in $\mathcal{F}_2$ or not for the resulting delta-matroid that is not further decomposed containing $\{b\}$. A similar situation arises for the case of a direct product of M_1 and M_2 where we still choose to decompose using elements a, b. In that case, for elements $A \cup B \in \mathcal{F}$, we must always include $B \in \mathcal{F}_2$ and possibly $B \cup \{b\} \in \mathcal{F}_2$, or always include $B \cup \{b\} \in \mathcal{F}_2$ and possibly $B \in \mathcal{F}_2$.

For the delta-matroids $M_i = (E_i, \mathcal{F}_i)$ that are not decomposed and must be zebra delta-matroids, we may choose what possibly included elements for $\mathcal{F}_i$ to exclude by choosing the corresponding $0 \leq r_i \leq s_i \leq |E_i|$ so as to satisfy the definition of a zebra delta-matroid, thus having to exclude all sets of size less than r_i or greater than s_i, while we may always choose to include sets of size from r_i to s_i.

There are thus d^k cases that take $O(c^k)$ time and reduce to a case for a smaller k, giving the recurrence $f(k) = d^k(c^k + f(k-1))$, $f(0) = 0$, on the time used for finding such a decomposition, that is time complexity of the order $O((cd)^{k^2})$. $\square$

A delta-matroid $M = (E, \mathcal{F})$ is *even* if for all $F, G \in \mathcal{F}$, $|F \Delta G|$ is even. We consider any class $\mathcal{C}$ of even delta-matroids, closed under restriction and direct sum, such that there is a polynomial time algorithm for delta-matroid parity on matroids in $\mathcal{C}$. Examples of $\mathcal{C}$ include even local delta-matroids [3] and linear delta-matroids over a given field [7].

A delta-matroid $M = (E, \mathcal{F})$ is a *$\mathcal{C}$-zebra delta-matroid* if for every feasible set $A \in \mathcal{F}$, there exists a delta-matroid $M_A = (E, \mathcal{F}_A)$ in $\mathcal{C}$ such that: (1) all sets $B \in \mathcal{F}$ such that $A \Delta B$ is of even size are also in $\mathcal{F}_A$; (2) if $B \in \mathcal{F} \cap \mathcal{F}_A$ and $C \in \mathcal{F}_A \setminus \mathcal{F}$ are such that $|B \Delta C| = 2$ and $|A \Delta C| = |A \Delta B| + 2$, then the two sets D such that $|B \Delta D| = |D \Delta C| = 1$ satisfy $C \in \mathcal{F}$.

We assume that a $\mathcal{C}$-zebra delta-matroid M is given together with appropriate presentations for the corresponding delta-matroids M_A in $\mathcal{C}$.

Theorem 7.7 *Suppose $M = (E, \mathcal{F})$ with $|E| = n$ is the direct sum of $\mathcal{C}$-zebra delta-matroids $M_i = (E_i, \mathcal{F}_i)$ with $|E_i| \leq k$, $|\mathcal{F}_i| \leq f$. Then delta-matroid parity on M, $\mathcal{L}$, can be solved in time polynomial in n and f.*

Proof. The proof is analogous to the proof of Theorem 7.2. As in Lemma 7.3, the problem reduces to the case where we have a feasible F and only a single $\{a, b\} \in \mathcal{L}$ such that $\{a, b\} \notin \mathcal{L}_F$, and one of the M_i has $E_i = \{b\}$.

Consider the problem in this form with $\mathcal{C}$-zebra delta-matroids $M_i = (E_i, \mathcal{F}_i)$. For $A = F \cap E_i$, let $M_i' = (E_i, \mathcal{F}_i)$ be the delta-matroid $(M_i)_A$ in $\mathcal{C}$ in the definition of $\mathcal{C}$-zebra delta-matroids. As in Lemma 7.4, the problem reduces to a problem where all but one of the M_i have been replaced by M_i'. The key point as before is that given an augmenting path be $a = x_1, \ldots, x_{2t+1}$ with $x_{2t+1} \in E_0$ in M', either this is also an augmenting path for M, or we have $F \Delta \{x_1, \ldots, x_{2j-2}\}$ feasible for M and for M', but $F \Delta \{x_1, \ldots, x_{2j}\}$ feasible for M' and not feasible for M, and then $F \Delta \{x_1, \ldots, x_{2j-1}\}$ is feasible for M by the definition of $\mathcal{C}$-zebra delta-matroids, giving an augmenting path for M as well. Finally, as in Lemma 7.5, we replace $M_0 = (E_0, \mathcal{F}_0)$ with $M_0' = (E_0, \{F_0\})$ for each $F_0 \in \mathcal{F}_0$ and find augmenting paths starting at a and ending in M_0' so that in the end, we will either have $F \cap E_0 = F_0$ or have found a shorter augmenting path, using the algorithm for delta-matroid parity in $\mathcal{C}$, since any algorithm for delta-matroid parity can be used to find an augmenting path starting at a if it exists. $\square$

An even delta-matroid $M = (E, \mathcal{F})$ is *local* if for every $F \in \mathcal{F}$ and every pairing $\mathcal{L}$, if $x_1, \ldots, x_{2k}$ is a path such that $F \Delta \{x_{2i-1}, x_{2i}\} \in \mathcal{F}$ for $2 \leq 2i \leq 2k$ and $\{x_{2i}, x_{2i+1}\} \in \mathcal{L}$ for $2 \leq 2i < 2k$, and there is no shorter path $x_1 = y_1, \ldots, y_{2l} = x_{2k}$ satisfying this property with $\{y_1, \ldots, y_{2l}\} \subset \{x_1, \ldots, x_{2k}\}$, then $F \Delta \{x_1, \ldots, x_{2k}\} \in \mathcal{F}$.

The algorithm of Dalmau and Ford [3] for local delta-matroid parity only requires in the case of even delta-matroids that this property hold, in an augmenting phase started at a feasible set $F \in \mathcal{F}$, for that particular feasible set F. We say that in that case M is F-local. The algorithm uses the fact that if $z_1, \ldots, z_{2r}$ is an F-augmenting, $\mathcal{L}_F$-alternating path, then by repeated application of Wenzel's strong exchange axiom for even delta-matroids, there exists a $\mathcal{L}_F$-alternating path $z_1 = x_1, \ldots, x_{2k} = z_{2r}$ such that $F \Delta \{x_{2i-1}, x_{2i}\} \in \mathcal{F}$ for $2 \leq 2i \leq 2k$. Such a path may be found by an augmentation in graph matching, by considering the graph consisting of all edges $\{x, y\}$ such that $F \Delta \{x, y\} \in \mathcal{F}$, plus edges $\{x, y\}$ in the given matching for $\{x, y\} \in \mathcal{L}_F$. The algorithm also attempts to find a shorter augmenting path, in the subgraph induced by $x_1, \ldots, x_{2i-1}, x_{2i+2}, \ldots, x_{2k}$, for each $2 \leq 2i < 2k$. When such a shorter augmenting path is not found, the definition of F-local guarantees that $x_1, \dots, x_{2k}$ is an F-augmenting path.

We generalize local-zebra delta-matroids by only requiring M_A to be A-local instead of local, and still obtain a polynomial time algorithm for the corresponding A-local-zebra delta-matroid parity problem. We thus have polynomial time algorithms when the even delta-matroids M_A in the definition of $\mathcal{C}$-zebra are all linear over a given field, or all A-local. These classes are closed under direct sums. More generally, we may allow M_A to be obtained from M_B in the class of delta-matroids linear over a given field or the class of B-local delta-matroids by contracting $\mathcal{K}$ such that $\mathcal{K}_B = \mathcal{K}$ and A consists of the elements of B not in pairs from $\mathcal{K}$.

A main example of local-zebra delta-matroids is the class of delta-matroids $M = (E, \mathcal{F})$ that satisfy a stronger exchange property, namely that for all feasible sets A, B and every element $x \in A\Delta B$, either $A\Delta\{x\}$ is feasible, or for every $y \in A\Delta\{x\}\Delta B$ the set $A\Delta\{x, y\}$ is feasible. In this case the local delta-matroid $M_A = (E, \mathcal{F}_A)$ can be defined by letting $\mathcal{F}_A$ be the set of all $B \subseteq E$ such that $A\Delta B$ is of even size and $A\Delta B \subseteq A\Delta D$ for some $D \in \mathcal{F}$.

Indeed, condition (2) in the definition of local-zebra holds by the stronger exchange property applied to B and D such that $A\Delta C \subseteq A\Delta D$. It remains to show that M_A is a local delta-matroid. The subsets $D \in \mathcal{F}$ can be chosen without loss of generality with $A\Delta D$ of maximum size over $D \in \mathcal{F}$. Any two such D must satisfy $|D_1\Delta D_2| = 2$, otherwise there exists $\{x, y\} \subset D_1\Delta D_2$ such that $|A\Delta(D_1\Delta\{x, y\})| = |A\Delta D_1| + 2$, and so $D_1\Delta\{x\}, D_1\Delta\{x, y\} \notin \mathcal{F}$ contrary to the stronger exchange property for D_1, D_2. There exists thus a set E such that $|D_i\Delta E| = 1$ for all such D_i, and if $|A\Delta D_i| > 1$ then E can be chosen so that $|A\Delta E| = |A\Delta D_i| + 1$. Furthermore, if there are at least two such feasible D_i, say D_1 and D_2, then we may not have two D_i such that $|A\Delta E| = |A\Delta D_i| + 1$ that are not feasible, say D_3 and D_4. Otherwise, if $D_i = E\Delta\{x_i\}$, then the possible intersections of $A\Delta F$ for $F \in \mathcal{F}$ with $X = \{x_1, x_3, x_4\}$ include $\emptyset$ and X, but do not include any other subset Y containing x_1 corresponding to some feasible G, by the stronger exchange property applied to D_1, G and x_1, contradicting M beign a delta-matroid by the exchange property for $\emptyset$, X and x_1.

Letting $E = D$ if there is only one such D_i, and letting $E = \emptyset$ if $|A\Delta D_i| \leq 1$, we have that $M_A = (E, \mathcal{F}_A)$ has $\mathcal{F}_A$ consisting of all sets G such that $A\Delta G$ is of even size and a subset of $A\Delta E$, or consisting of all these sets G except for a single G with $|A\Delta G| = |A\Delta E| - 1$. Suppose now $G \in \mathcal{F}_A$ and some path $x_1, \dots, x_{2k}$ fails to satisfy the definition of G-local for M_A. If $k \geq 3$, then either $G\Delta\{x_1, x_4\}$ or $G\Delta\{x_1, x_6\}$ is in $\mathcal{F}_A$ since at most one even size subset of $G\Delta E$ fails to give a feasible set. Thus $x_1, \dots, x_{2k}$ has a shortcut path as in the definition of G-local when $k \geq 3$. If $k = 2$, then either $G\Delta\{x_1, x_4\} \in \mathcal{F}_A$ giving again a shortcut, or $G\Delta\{x_1, x_2, x_3, x_4\} \in \mathcal{F}_A$ because at most one even size subset of $G\Delta E$ is missing, satisfying the definition of G-local. Thus M_A is G-local for all $G \in \mathcal{F}_A$, that is, M_A is local and therefore M is local-zebra.

8 Conclusions

We considered bipartite Boolean constraint satisfaction having two different sets of allowed constraint types for both sides of the bipartition. This case is properly bipartite only if at least one side does not contain equality. We obtain a classification for these problems if they do not have equality in both sides. All the algorithms work even in an oracle model, leaving open when equality is an allowed constraint in both sides cases of delta-matroid parity, which cannot in general be solved in oracle model. We also obtain a full classification for k-partite Boolean constraint satisfaction with $k \geq 3$.

Known polynomial cases of delta-matroid parity include local and linear delta-matroids, and delta-matroids obtained from these by simulation. All of the remaining known polynomial cases are covered by cases studied in this paper. The first case is that of delta-matroids that do not contain inequality. The second case is obtained from any of the known polynomial cases $\mathcal{C}$ of even delta-matroids by considering the corresponding $\mathcal{C}$-zebra delta-matroids, and adding closure under simulation as well. Here $\mathcal{C}$ may be the class of delta-matroids obtained by simulation from linear delta-matroids over a given field, or the class of delta-matroids obtained by simulation from A-local delta-matroids.

References

[1] A. Bouchet and B. Jackson, Parity systems and the delta-matroid intersection problem, The Electronic Journal of Combinatorics 7 (2000).

[2] G.P. Cornuejols, General factors of graphs, J. Combinatorial Theory B 45 (1988) 185–198.

[3] V. Dalmau and D. Ford, Generalized satisfiability with k occurrences per variable: a study through delta-matroid parity, Proc. 28th International Symposium of Mathematical Foundations of Computer Science (2003).

[4] J. Edmonds, Matroid intersection, Annals of Discrete Math., 14 (1979) 39–49.

[5] T. Feder, Fanout limitations on constraint systems, Theoretical Computer Science 255 (2001) 281–293.

[6] H.N. Gabow and M. Stallman, An augmenting path algorithm for linear matroid parity, Combinatorica 6 (1986) 123–150.

[7] J.F. Geelen, S. Iwata, and K. Murota, The linear delta-matroid parity problem, J. Comb. Theory B 88 (2003), 377–398.

[8] G. Istrate, Looking for a version of Schaefer's dichotomy theorem when each variable occurs at most twice, TR652, Computer Science Dept., U. Rochester (1997).

[9] L. Lovász, Matroid matching and some applications, J. Combinatorial Theory Ser. B 28 (1980) 208–236.

[10] L. Lovász and M. Plummer, Matching Theory, North-Holland, 1986.

[11] T. J. Schaefer, The complexity of satisfiability problems, Proc. 10th ACM Symp. on Theory of Computing (1978) 216–226.

[12] W. Wenzel, δ-matroids with the strong exchange conditions, Appl. Math. Lett. 6 (1993) 67–70.

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'