The Connectivity of Boolean Satisfiability: Computational and Structural Dichotomies

P. Gopalan[1], Ph.G. Kolaitis[2⋆], E.N. Maneva[3], and C.H. Papadimitriou[3]

[1] Georgia Tech.
[2] IBM Almaden Research Center
[3] UC Berkeley

Abstract. Given a Boolean formula, do its solutions form a connected subgraph of the hypercube? This and other related connectivity considerations underlie recent work on random Boolean satisfiability. We study connectivity properties of the space of solutions of Boolean formulas, and establish computational and structural dichotomies. Specifically, we first establish a dichotomy theorem for the complexity of the st-connectivity problem for Boolean formulas in Schaefer's framework. Our result asserts that the tractable side is more generous than the tractable side in Schaefer's dichotomy theorem for satisfiability, while the intractable side is PSPACE-complete. For the connectivity problem, we establish a dichotomy along the same boundary between membership in coNP and PSPACE-completeness. Furthermore, we establish a structural dichotomy theorem for the diameter of the connected components of the solution space: for the PSPACE-complete cases, the diameter can be exponential, but in all other cases it is linear. Thus, small diameter and tractability of the st-connectivity problem are remarkably aligned.

1 Introduction

In 1978, T.J. Schaefer [1] introduced a rich framework for expressing variants of Boolean satisfiability and proved a remarkable *dichotomy theorem*: the satisfiability problem is in P for certain classes of Boolean formulas, while it is NP-complete for all other classes in the framework. In a single stroke, this result pinpoints the computational complexity of all well-known variants of Sat, such as 3-Sat, Horn 3-Sat, Not-All-Equal 3-Sat, and 1-in-3 Sat. Schaefer's work paved the way for a series of investigations establishing dichotomies for several aspects of satisfiability, including optimization [2–4], counting [5], inverse satisfiability [6], minimal satisfiability [7], 3-valued satisfiability [8] and propositional abduction [9].

Our aim in this paper is to carry out a comprehensive exploration of a different aspect of Boolean satisfiability, namely, the *c*onnectivity properties of the space of solutions of Boolean formulas. The solutions (satisfying assignments) of a given n-variable Boolean formula φ induce a subgraph $G(\varphi)$ of the

⋆ On leave from UC Santa Cruz

n-dimensional hypercube. Thus, the following two decision problems, called the *connectivity problem* and the *st-connectivity problem*, arise naturally: (i) Given a Boolean formula φ, is $G(\varphi)$ connected? (ii) Given a Boolean formula φ and two solutions $\mathbf{s}$ and $\mathbf{t}$ of φ, is there a path from $\mathbf{s}$ to $\mathbf{t}$ in $G(\varphi)$?

We believe that connectivity properties of Boolean satisfiability merit study in their own right, as they shed light on the structure of the solution space of Boolean formulas. Moreover, in recent years the structure of the space of solutions for random instances has been the main consideration at the basis of both algorithms for and mathematical analysis of the satisfiability problem [10–13]. It has been conjectured for 3-SAT [12] and proved for 8-SAT [14, 15], that the solution space fractures as one approaches the *critical region* from below. This apparently leads to performance deterioration of the standard satisfiability algorithms, such as WalkSAT [16] and DPLL [17]. It is also the main consideration behind the design of the survey propagation algorithm, which has far superior performance on random instances of satisfiability [12]. This body of work has served as a motivation to us for pursuing the investigation reported here. While there has been an intensive study of the structure of the solution space of Boolean satisfiability problems for random instances, our work seems to be the first to explore this issue from a worst-case viewpoint.

Our first main result is a dichotomy theorem for the *st*-connectivity problem. This result reveals that the tractable side is much more generous than the tractable side for satisfiability, while the intractable side is PSPACE-complete. Specifically, Schaefer showed that the satisfiability problem is solvable in polynomial time precisely for formulas built from Boolean relations all of which are bijunctive, or all of which are Horn, or all of which are dual Horn, or all of which are affine. We identify new classes of Boolean relations, called *tight* relations, that properly contain the classes of bijunctive, Horn, dual Horn, and affine relations. We show that *st*-connectivity is solvable in linear time for formulas built from tight relations, and PSPACE-complete in all other cases. Our second main result is a dichotomy theorem for the connectivity problem: it is in coNP for formulas built from tight relations, and PSPACE-complete in all other cases.

In addition to these two complexity-theoretic dichotomies, we establish a structural dichotomy theorem for the diameter of the connected components of the solution space of Boolean formulas. This result asserts that, in the PSPACE-complete cases, the diameter of the connected components can be exponential, but in all other cases it is linear. Thus, small diameter and tractability of the *st*-connectivity problem are remarkably aligned.

To establish these results, we first show that all *tight* relations have "good" structural properties. Specifically, in a tight relation every component has a unique minimum element, or every component has a unique maximum element, or the Hamming distance coincides with the shortest-path distance in the relation. These properties are inherited by every formula built from tight relations, and yield both small diameter and linear algorithms for *st*-connectivity.

Next, the challenge is to show that for non-tight relations, both the connectivity problem and the *st*-connectivity problem are PSPACE-hard. In Schaefer's

Dichotomy Theorem, NP-hardness of satisfiability was a consequence of an *expressibility* theorem, which asserted that every Boolean relation can be obtained as a projection over a formula built from clauses in the "hard" relations. Schaefer's notion of expressibility is inadequate for our problem. Instead, we introduce and work with a delicate and more strict notion of expressibility, which we call *faithful expressibility*. Intuitively, faithful expressibility means that, in addition to definability via a projection, the space of witnesses of the existential quantifiers in the projection has certain strong connectivity properties that allow us to capture the graph structure of the relation that is being defined. It should be noted that Schaefer's Dichotomy Theorem can also be proved using a Galois connection and Post's celebrated classification of the lattice of Boolean clones (see [18]). This method, however, does not appear to apply to connectivity, as the boundaries discovered here cut across Boolean clones. Thus, the use of faithful expressibility or some other refined definability technique seems unavoidable.

The first step towards proving PSPACE-completeness is to show that both connectivity and st-connectivity are hard for 3-CNF formulae; this is proved by a reduction from a generic PSPACE computation. Next, we identify the simplest relations that are not tight: these are ternary relations whose graph is a path of length 4 between assignments at Hamming distance 2. We show that these paths can faithfully express all 3-CNF clauses. The crux of our hardness result is an *expressibility* theorem to the effect that one can faithfully express such a path from any set of relations which is not tight.

Our original hope was that tractability results for connectivity could conceivably inform heuristic algorithms for satisfiability and enhance their effectiveness. In terms of this motivation, our findings are *prima facie* negative: we show that when satisfiability is intractable, then connectivity is also intractable. But our results do contain a glimmer of hope: we show that there are broad classes of intractable satisfiability problems, those built from tight relations, with polynomial *st*-connectivity and small diameter. It would be interesting to investigate if these properties make random instances built from tight relations easier for WalkSAT and similar heuristics, and if so, whether such heuristics are amenable to rigorous analysis.

For want of space, some proofs, as well as some additional results, are omitted here; they can be found in the full version available at ECCC.

2 Basic Concepts and Statements of Results

A *logical relation* R is a non-empty subset of $\{0,1\}^k$, for some $k \geq 1$; k is the *arity* of R. Let S be a finite set of logical relations. A CNF(S)*-formula* over a set of variables $V = \{x_1, \ldots, x_n\}$ is a finite conjunction $C_1 \wedge \ldots \wedge C_n$ of clauses built using relations from S, variables from V, and the constants 0 and 1; this means that each C_i is an expression of the form $R(\xi_1, \ldots, \xi_k)$, where $R \in S$ is a relation of arity k, and each ξ_j is a variable in V or one of the constants 0, 1.

The *satisfiability problem* $\textsc{Sat}(S)$ associated with a finite set S of logical relations asks: given a CNF(S)-formula φ, is it satisfiable? All well known restrictions of Boolean satisfiability, such as 3-Sat, Not-All-Equal 3-Sat, and Positive 1-in-3 Sat, can be cast as $\textsc{Sat}(S)$ problems, for a suitable choice of S. For instance, POSITIVE 1-IN-3 SAT is SAT($\{R_{1/3}\}$), where $R_{1/3} = \{100, 010, 001\}$. Schaefer [1] identified the complexity of *every* satisfiability problem SAT(S). To state Schaefer's main result, we need to define some basic concepts.

Definition 1 Let R be a logical relation.
(1) R is *bijunctive* if it is the set of solutions of a 2-CNF formula.

(2) R is *Horn* if it is the set of solutions of a Horn formula, where a Horn formula is a CNF formula such that each conjunct has at most one positive literal.

(3) R is *dual Horn* if it is the set of solutions of a dual Horn formula, where a dual Horn formula is a CNF formula such that each conjunct has at most one negative literal.

(4) R is *affine* if it is the set of solutions of a system of linear equations over $\mathbb{Z}_2$.

Each of these types of logical relations can be characterized in terms of *closure* properties [1]. A relation R is bijunctive if and only if it is closed under the *majority* operation (if $\mathbf{a}, \mathbf{b}, \mathbf{c} \in R$, then $\mathrm{maj}(\mathbf{a}, \mathbf{b}, \mathbf{c}) \in R$, where $\mathrm{maj}(\mathbf{a}, \mathbf{b}, \mathbf{c})$ of $\mathbf{a}, \mathbf{b}, \mathbf{c}$ is the vector whose i-th bit is the majority of a_i, b_i, c_i). A relation R is Horn if and only if it is closed under $\vee$ (if $\mathbf{a}, \mathbf{b} \in R$, then $\mathbf{a} \vee \mathbf{b} \in R$, where, $\mathbf{a} \vee \mathbf{b}$ is the vector whose i-th bit is $a_i \vee b_i$). Similarly, R is dual Horn if and only if it is closed under $\wedge$. Finally, R is affine if and only if it is closed under $\mathbf{a} \oplus \mathbf{b} \oplus \mathbf{c}$.

Definition 2 A set S of logical relations is *Schaefer* if at least one of the following holds: (1) Every relation in S is bijunctive; (2) Every relation in S is Horn; (3) Every relation in S is dual Horn; (4) Every relation in S is affine.

Theorem 1 (Schaefer's Dichotomy Theorem [1]) *If S is Schaefer, then* SAT(S) *is in* P*; otherwise,* SAT(S) *is* NP*-complete.*

Note that the closure properties of Schaefer sets yield a cubic algorithm for determining, given a finite set S of relations, whether SAT(S) is in P or NP-complete (the input size is the sum of the sizes of relations in S).

Here, we are interested in the connectivity properties of the space of solutions of CNF(S)-formulas. If φ is a CNF(S)-formula with n variables, then $G(\varphi)$ denotes the subgraph of the n-dimensional hypercube induced by the solutions of φ. Thus, the vertices of $G(\varphi)$ are the solutions of φ, and there is an edge between two solutions of $G(\varphi)$ precisely when they differ in a single variable. We consider the following two algorithmic problems for CNF(S)-formulas.

(1) The *st-connectivity* problem ST-CONN(S): given a CNF(S)-formula φ and two solutions $\mathbf{s}$ and $\mathbf{t}$ of φ, is there a path from $\mathbf{s}$ to $\mathbf{t}$ in $G(\varphi)$?
(2) The *connectivity* problem CONN(S): given a CNF(S)-formula φ, is $G(\varphi)$ connected?

To pinpoint the computational complexity of ST-CONN(S) and CONN(S), we need to introduce certain new types of relations.

Definition 3 Let $R \subseteq \{0, 1\}^k$ be a logical relation.

(1) R is *componentwise bijunctive* if every connected component of $G(R)$ is bijunctive.

(2) R is OR-*free* if the relation $\mathrm{OR} = \{01, 10, 11\}$ cannot be obtained from R by setting $k-2$ of the coordinates of R to a constant $\mathbf{c} \in \{0,1\}^{k-2}$. In other words, R is OR-free if $(x_1 \vee x_2)$ is not definable from R by fixing $k-2$ variables.
(3) R is NAND-*free* if $(\bar{x}_1 \vee \bar{x}_2)$ is not definable from R by fixing $k-2$ variables.

The next lemma is proved using the closure properties of bijunctive, Horn, and dual Horn relations. (We skip the easy proof).

Lemma 1 *Let R be a logical relation.*
(1) *If R is bijunctive, then R is componentwise bijunctive.*
(2) *If R is Horn, then R is* OR-*free.*
(3) *If R is dual Horn, then R is* NAND-*free.*
(4) *If R is affine, then R is componentwise bijunctive,* OR-*free, and* NAND-*free.*

These containments are proper. For instance, $R_{1/3} = \{100, 010, 001\}$ is componentwise bijunctive, but not bijunctive as $\mathrm{maj}(100, 010, 001) = 000 \notin R_{1/3}$.
We are now ready to introduce the key concept of a *tight* set of relations.

Definition 4 A set S of logical relations is *tight* if at least one of the following three conditions holds: (1) Every relation in S is componentwise bijunctive; (2) Every relation in S is OR-free; (3) Every relation in S is NAND-free.

In view of Lemma 1, if S is Schaefer, then it is tight. The converse, however, does not hold. It is also easy to see that there is a polynomial-time algorithm for testing whether a given finite set S of logical relations is tight. Our first main result is a dichotomy theorem for the computational complexity of st-Conn(S).

Theorem 2 *Let S be a finite set of logical relations. If S is tight, then* st-Conn(S) *is in* P*; otherwise,* Conn(S) *is* PSPACE-*complete.*

Our second main result asserts that the dichotomy in the computational complexity of st-Conn(S) is accompanied by a parallel structural dichotomy in the size of the diameter of $G(\varphi)$ (where, for a CNF(S)-formula φ, the *diameter of* $G(\varphi)$ is the maximum of the diameters of the components of $G(\varphi)$).

Theorem 3 *Let S be a finite set of logical relations. If S is tight, then for every* CNF(S)*-formula φ, the diameter of $G(\varphi)$ is linear in the number of variables of φ; otherwise, there are* CNF(S)*-formulas φ such that the diameter of $G(\varphi)$ is exponential in the number of variables of φ.*

Our third main result establishes a dichotomy for the complexity of Conn(S).

Theorem 4 *Let S be a finite set of logical relations. If S is tight, then* Conn(S) *is in* coNP*; otherwise, it is* PSPACE-*complete.*

We also show that if S is tight, but not Schaefer, then Conn(S) is coNP-complete. Our results and their comparison to Schaefer's Dichotomy Theorem are summarized in the table below.

S	Sat(S)	st-Conn(S)	Conn(S)	Diameter
Schaefer	in P	in P	in coNP	$O(n)$
Tight, not Schaefer	NP-compl.	in P	coNP-compl.	$O(n)$
Not tight	NP-compl.	PSPACE-compl.	PSPACE-compl.	$2^{\Omega(\sqrt{n})}$

As an application, the set $S = \{R_{1/3}\}$, where $R_{1/3} = \{100, 010, 001\}$, is tight, but not Schaefer. It follows that Sat(S) is NP-complete (recall that this problem is Positive 1-in-3 Sat), st-Conn(S) is in P, and Conn(S) is coNP-complete. Consider also the set $S = \{R_{\text{NAE}}\}$, where $R_{\text{NAE}} = \{0,1\}^3 \setminus \{000, 111\}$. This set is not tight, hence Sat(S) is NP-complete (this problem is Positive Not-All-Equal 3-Sat), while both st-Conn(S) and Conn(S) are PSPACE-complete.

We conjecture that if S is Schaefer, then Conn(S) is in P. If this conjecture is true, it will follow that the complexity of Conn(S) exhibits a *trichotomy*: if S is Schaefer, then Conn(S) is in P; if S is tight, but not Schaefer, then Conn(S) is coNP-complete; if S is not tight, then Conn(S) is PSPACE-complete.

3 The Easy Cases of Connectivity

In this section, we determine the complexity of Conn(S) and st-Conn(S) for tight sets S of logical relations, and also show that for such sets, the diameter of $G(\varphi)$ of CNF(S)-formula φ is linear. We prove only the key structural properties of tight relations here, and defer the rest to the full version.

We will use $\mathbf{a}, \mathbf{b}, \ldots$ to denote Boolean vectors, and $\mathbf{x}$ and $\mathbf{y}$ to denote vectors of variables. We write $|\mathbf{a}|$ to denote the Hamming weight (number of 1's) of a Boolean vector $\mathbf{a}$. Given two Boolean vectors $\mathbf{a}$ and $\mathbf{b}$, we write $|\mathbf{a}-\mathbf{b}|$ to denote the Hamming distance between $\mathbf{a}$ and $\mathbf{b}$. Finally, if $\mathbf{a}$ and $\mathbf{b}$ are solutions of a Boolean formula φ and lie in the same component of $G(\varphi)$, then we write $d_\varphi(\mathbf{a}, \mathbf{b})$ to denote the shortest-path distance between $\mathbf{a}$ and $\mathbf{b}$ in $G(\varphi)$.

3.1 The st-Conn Problem for Tight Sets

Lemma 2 *Let S be a set of componentwise bijunctive relations and φ a* CNF(S)*-formula. If $\mathbf{a}$ and $\mathbf{b}$ are two solutions of φ that lie in the same component of $G(\varphi)$, then $d_\varphi(\mathbf{a}, \mathbf{b}) = |\mathbf{a} - \mathbf{b}|$.*

Proof. (Sketch) Consider first the special case in which every relation in S is bijunctive. In this case, φ is equivalent to a 2-CNF formula and so the space of solutions of φ is closed under maj. We show that there is a path in $G(\varphi)$ from $\mathbf{a}$ to $\mathbf{b}$, such that along the path only the assignments on variables with indices from the set $D = \{i : a_i \neq b_i\}$ change. This implies that the shortest path is of length $|D|$ by induction on $|D|$. Consider any path $\mathbf{a} \to \mathbf{u^1} \to \cdots \to \mathbf{u^r} \to \mathbf{b}$ in $G(\varphi)$. We construct another path by replacing $\mathbf{u^i}$ by $\mathbf{v^i} = \text{maj}\ (\mathbf{a}, \mathbf{u^i}, \mathbf{b})$ for $i = 1, \ldots, r$, and removing repetitions. This path has the desired property.

For the general case, it can be shown that every component F of $G(\varphi)$ is the solution space of a 2-CNF formula φ'. If C is a clause of φ involving a relation R

in S, then the projection of F on the variables of C is contained in a component of R. Then the formula φ' is obtained from φ as follows: replace each clause C of φ by a 2-CNF formula expressing the component of R that contains the projection of F on the variables of C. $\square$

Corollary 1 *Let S be a set of componentwise bijunctive relations. Then (1) for every $\varphi \in \mathrm{CNF}(S)$ with n variables, the diameter of each component of $G(\varphi)$ is bounded by n; (2)* st-Conn(S) *is in* P*; and (3)* Conn(S) *is in* coNP.

Next, we consider sets of OR-free relations (sets of NAND-free relations are handled dually). Define the *coordinate-wise partial order* $\leq$ on Boolean vectors as follows: $\mathbf{a} \leq \mathbf{b}$ if $a_i \leq b_i$, for each i.

Lemma 3 *Let S be a set of* OR*-free relations and φ a* CNF(S)*-formula. Every component of $G(\varphi)$ contains a minimum solution with respect to the coordinate-wise order; moreover, every solution is connected to the minimum solution in the same component via a monotone path.*

Proof. Suppose there are two distinct minimal assignments $\mathbf{u}$ and $\mathbf{u'}$ in some component of $G(\varphi)$. Consider the path between them where the maximum Hamming weight of assignments on the path is minimized. If there are many such paths, pick one where the smallest number of assignments have the maximum Hamming weight. Denote this path by $\mathbf{u} = \mathbf{u^1} \to \mathbf{u^2} \cdots \to \mathbf{u^r} = \mathbf{u'}$. Let $\mathbf{u^i}$ be the assignment of largest Hamming weight in the path. Then $\mathbf{u^i} \neq \mathbf{u}$ and $\mathbf{u^i} \neq \mathbf{u'}$, since $\mathbf{u}$ and $\mathbf{u'}$ are minimal. The assignments $\mathbf{u^{i-1}}$ and $\mathbf{u^{i+1}}$ differ in exactly 2 variables, say, in x_1 and x_2. So $\{u_1^{i-1}u_2^{i-1},\ u_1^i u_2^i,\ u_1^{i+1}u_2^{i+1}\} = \{01, 11, 10\}$. Let $\hat{\mathbf{u}}$ be such that $\hat{u}_1 = \hat{u}_2 = 0$, and $\hat{u}_i = u_i$ for $i > 2$. If $\hat{\mathbf{u}}$ is a solution, then the path $\mathbf{u^1} \to \mathbf{u^2} \to \cdots \to \mathbf{u^i} \to \hat{\mathbf{u}} \to \mathbf{u^{i+1}} \to \cdots \to \mathbf{u^r}$ contradicts the way we chose the original path. Therefore, $\hat{\mathbf{u}}$ is not a solution. This means that there is a clause that is violated by it, but is satisfied by $\mathbf{u^{i-1}}$, $\mathbf{u^i}$, and $\mathbf{u^{i+1}}$. So the relation corresponding to that clause is not OR-free, which is a contradiction.

The unique minimal solution in a component is its minimum solution. Furthermore, starting from any assignment $\mathbf{s}$ in the component, and repeatedly flipping variables from 1 to 0 provides a monotone path to the minimum. $\square$

Corollary 2 *Let S be a set of* OR*-free relations. Then (1) For every $\varphi \in$* CNF(S) *with n variables, the diameter of each component of $G(\varphi)$ is bounded by $2n$; (2)* st-Conn(S) *is in* P*; and (3)* Conn(S) *is in* coNP.

4 The PSPACE-Complete Cases of Connectivity

If $k \geq 2$, then a *k-clause* is a disjunction of k variables or negated variables. For $0 \leq i \leq k$, let D_i be the set of all satisfying truth assignments of the k-clause whose first i literals are negated, and let $S_k = \{D_0, D_1, \ldots, D_k\}$. Thus, CNF$(S_k)$ is the collection of k-CNFformulas.

The starting point of the proof is to show that Conn(S_3) and st-Conn(S_3) are PSPACE-complete. The proof is fairly intricate, and is via a direct reduction from the computation of a polynomial-space Turing machine. We also show that 3-CNF formulas can have exponential diameter, by inductively constructing a path of length at least $2^{\frac{n}{2}}$ on n variables and then identifying it with the solution space of a 3-CNF formula with $O(n^2)$ clauses.

Lemma 4 *st-*$\textsc{Conn}(S_3)$ *and* $\textsc{Conn}(S_3)$ *are* PSPACE*-complete.*

Lemma 5 *For n even, there is a* 3-CNF *formula φ_n with n variables and $O(n^2)$ clauses, such that $G(\varphi_n)$ is a path of length greater than $2^{\frac{n}{2}}$.*

4.1 Faithful Expressibility

From here onwards, all our hardness results are proved by showing that if S is a non-tight set, then every 3-clause is expressible from S in a certain special way that we describe next. In his dichotomy theorem, Schaefer [1] used the following notion of expressibility: a relation R is *expressible from* a set S of relations if there is a CNF(S)-formula φ so that $R(\mathbf{x}) \equiv \exists \mathbf{y}\ \varphi(\mathbf{x}, \mathbf{y})$. This notion, is not sufficient for our purposes. Instead, we introduce a more delicate notion, which we call *faithful expressibility.* Intuitively, we view the relation R as a subgraph of the hypercube, rather than just a subset, and require that this graph structure be also captured by the formula φ.

Definition 5 *A relation R is* faithfully expressible *from a set of relations S if there is a* CNF(S)*-formula φ such that:*

(1) *$R = \{\mathbf{a} : \ \exists \mathbf{y}\ \varphi(\mathbf{a}, \mathbf{y})\}$;*
(2) *For every $\mathbf{a} \in R$, the graph $G(\varphi(\mathbf{a}, \mathbf{y}))$ is connected;*
(3) *For $\mathbf{a}, \mathbf{b} \in R$ with $|\mathbf{a} - \mathbf{b}| = 1$, there exists a $\mathbf{w}$ such that $(\mathbf{a}, \mathbf{w})$ and $(\mathbf{b}, \mathbf{w})$ are solutions of φ.*

For $\mathbf{a} \in R$, the *witnesses* of $\mathbf{a}$ are the $\mathbf{y}$'s such that $\varphi(\mathbf{a}, \mathbf{y})$. The last two conditions say that the witnesses of $\mathbf{a} \in R$ are connected, and that neighboring $\mathbf{a}, \mathbf{b} \in R$ have a common witness. This allows us to simulate an edge $(\mathbf{a}, \mathbf{b})$ in $G(R)$ by a path in $G(\varphi)$, and thus relate the connectivity properties of the solution spaces. There is however, a price to pay: it is much harder to come up with formulas that faithfully express a relation R. An example is when S is the set of all paths of length 4 in $\{0, 1\}^3$, a set that plays a crucial role in our proof. While S_3 is easily expressible from S in Schaefer's sense, the CNF(S)-formulas that faithfully express S_3 are fairly complicated and have a large witness space.

Lemma 6 *Let S and S' be sets of relations such that every $R \in S$ is faithfully expressible from S'. Given a* CNF(S)*-formula $\psi(\mathbf{x})$, one can efficiently construct a* CNF(S')*-formula $\varphi(\mathbf{x}, \mathbf{y})$ such that:*

(1) *$\psi(\mathbf{x}) \equiv \exists \mathbf{y}\ \varphi(\mathbf{x}, \mathbf{y})$;*
(2) *if $(\mathbf{s}, \mathbf{w}^{\mathbf{s}}), (\mathbf{t}, \mathbf{w}^{\mathbf{t}}) \in \varphi$ are connected in $G(\varphi)$ by a path of length d, then there is a path from $\mathbf{s}$ to $\mathbf{t}$ in $G(\psi)$ of length at most d;*
(3) *If $\mathbf{s}, \mathbf{t} \in \psi$ are connected in $G(\psi)$, then for every witness $\mathbf{w}^{\mathbf{s}}$ of $\mathbf{s}$, and every witness $\mathbf{w}^{\mathbf{t}}$ of $\mathbf{t}$, there is a path from $(\mathbf{s}, \mathbf{w}^{\mathbf{s}})$ to $(\mathbf{t}, \mathbf{w}^{\mathbf{t}})$ in $G(\varphi)$.*

Proof. Suppose ψ is a formula on n variables that consists of m clauses $C_1, \ldots, C_m$. For clause C_j, assume that the set of variables is $V_j \subseteq [n]$, and that it involves relation $R_j \in S$. Thus, $\psi(\mathbf{x})$ is $\wedge_{j=1}^m R_j(\mathbf{x}_{V_j})$. Let φ_j be the faithful expression for R_j from S', so that $R_j(\mathbf{x}_{V_j}) \equiv \exists \mathbf{y}_j\ \varphi_j(\mathbf{x}_{V_j}, \mathbf{y}_j)$. Let $\mathbf{y}$ be the vector $(\mathbf{y}_1, \ldots, \mathbf{y}_m)$ and let $\varphi(\mathbf{x}, \mathbf{y})$ be the formula $\wedge_{j=1}^m \varphi_j(\mathbf{x}_{V_j}, \mathbf{y}_j)$. Then $\psi(\mathbf{x}) \equiv \exists \mathbf{y}\ \varphi(\mathbf{x}, \mathbf{y})$.

Statement (2) follows from (1) by projection of the path on the coordinates of $\mathbf{x}$. For statement (3), consider $\mathbf{s}, \mathbf{t} \in \psi$ that are connected in $G(\psi)$ via a path $\mathbf{s} = \mathbf{u^0} \to \mathbf{u^1} \to \cdots \to \mathbf{u^r} = \mathbf{t}$. For every $\mathbf{u^i}, \mathbf{u^{i+1}}$, and clause C_j, there exists an assignment $\mathbf{w^i}_j$ to $\mathbf{y}_j$ such that both $(\mathbf{u^i}_{V_j}, \mathbf{w^i}_j)$ and $(\mathbf{u^{i+1}}_{V_j}, \mathbf{w^i}_j)$ are solutions of φ_j, by condition (2) of faithful expressibility. Thus $(\mathbf{u^i}, \mathbf{w^i})$ and $(\mathbf{u^{i+1}}, \mathbf{w^i})$ are both solutions of φ, where $\mathbf{w^i} = (\mathbf{w^i}_1, \ldots \mathbf{w^i}_m)$. Further, for every $\mathbf{u^i}$, the space of solutions of $\varphi(\mathbf{u^i}, \mathbf{y})$ is the product space of the solutions of $\varphi_j(\mathbf{u^i}_{V_j}, \mathbf{y}_j)$ over $j = 1, \ldots, m$. Since these are all connected by condition (3) of faithful expressibility, $G(\varphi(\mathbf{u^i}, \mathbf{y}))$ is connected. The following describes a path from $(\mathbf{s}, \mathbf{w^s})$ to $(\mathbf{t}, \mathbf{w^t})$ in $G(\varphi)$: $(\mathbf{s}, \mathbf{w^s}) \rightsquigarrow (\mathbf{s}, \mathbf{w^0}) \to (\mathbf{u^1}, \mathbf{w^0}) \rightsquigarrow (\mathbf{u^1}, \mathbf{w^1}) \to \cdots \rightsquigarrow (\mathbf{u^{r-1}}, \mathbf{w^{r-1}}) \to (\mathbf{t}, \mathbf{w^{r-1}}) \rightsquigarrow (\mathbf{t}, \mathbf{w^t})$. Here $\rightsquigarrow$ indicates a path in $G(\varphi(\mathbf{u^i}, \mathbf{y}))$. $\square$

Corollary 3 *Suppose S and S' are as in Lemma 6.*

(1) *There are polynomial time reductions from* Conn(S) *to* Conn(S')*, and from* st-Conn(S) *to* st-Conn(S')*.*

(2) *Given a* CNF(S)*-formula $\psi(\mathbf{x})$ with m clauses, one can efficiently construct a* CNF(S')*-formula $\varphi(\mathbf{x}, \mathbf{y})$ such that the length of $\mathbf{y}$ is $O(m)$ and the diameter of the solution space does not decrease.*

4.2 Expressing 3-clauses from non-tight Sets of Relations

In order prove Theorems 2, 3 and 4, it suffices to prove the following Lemma:

Lemma 7 *If set S of relations is non-tight, S_3 is faithfully expressible from S.*

First, observe that all 2-clauses are faithfully expressible from S. There exists $R \in S$ which is not OR-free, so we can express $(x_1 \vee x_2)$ by substituting constants in R. Similarly, we can express $(\bar{x}_1 \vee \bar{x}_2)$ using a relation that is not NAND-free. The last 2-clause $(x_1 \vee \bar{x}_2)$ can be obtained from OR and NAND by a technique that corresponds to reverse resolution. $(x_1 \vee \bar{x}_2) = \exists y\ (x_1 \vee y) \wedge (\bar{y} \vee \bar{x}_2)$. It is easy to see that this gives a faithful expression. From here onwards we assume that S contains all 2-clauses. The proof now proceeds in four steps.

Step 1: *Faithfully expressing a relation in which some distance expands.* For a relation R, we say that the distance between $\mathbf{a}$ and $\mathbf{b}$ *expands* if $\mathbf{a}$ and $\mathbf{b}$ are connected in $G(R)$, but $d_R(\mathbf{a}, \mathbf{b}) > |\mathbf{a} - \mathbf{b}|$. By Lemma 2 no distance expands in componentwise bijunctive relations. This property also holds for the relation $R_{\mathrm{NAE}} = \{0,1\}^3 \setminus \{000, 111\}$, which is not componentwise bijunctive. However, we show that if Q is not componentwise bijunctive, then, by adding 2-clauses, we can faithfully express a relation Q' in which some distance expands. For instance, when $Q = R_{\mathrm{NAE}}$, then we can take $Q'(x_1, x_2, x_3) = R_{\mathrm{NAE}}(x_1, x_2, x_3) \wedge (\bar{x_1} \vee \bar{x}_3)$. The distance between $\mathbf{a} = 100$ and $\mathbf{b} = 001$ in Q' expands. Similarly, in the

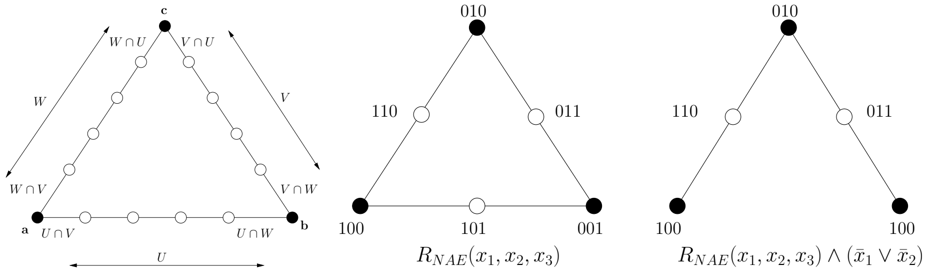

Fig. 1. Proof of Lemma 8

general construction, we identify **a** and **b** on a cycle, and add 2-clauses that eliminate all the vertices along the shorter arc between **a** and **b**.

Step 2: *Expressing a path of length $r + 2$ between assignments at distance r.* The relation Q' obtained in Step 1 may have several disconnected components. This *cleanup* step isolates a pair of assignments whose distance expands. By adding 2-clauses, we obtain a relation T that consists of a pair of assignments $\mathbf{a}, \mathbf{b}$ of Hamming distance r and a path of length $r + 2$ between them.

Step 3: *Faithfully expressing paths of length* 4.
Let P denote the set of all ternary relations whose graph is a path of length 4 between two assignments at Hamming distance 2. Up to permutations of coordinates, there are 6 such relations. Each of them is the conjunction of a 3-clause and a 2-clause. For instance, the relation $M = \{100, 110, 010, 011, 001\}$ can be written as of $(x_1 \vee x_2 \vee x_3) \wedge (\bar{x}_1 \vee \bar{x}_3)$. These relations are "minimal" examples of relations that are not componentwise bijunctive. By projecting out intermediate variables from the path T obtained in Step 2, we faithfully express one of the relations in P. We faithfully express other relations in P using this relation.

Step 4: *Faithfully expressing S_3.*
We faithfully express $(x_1 \vee x_2 \vee x_3)$ from M using a formula derived from a gadget in [19]. This gadget expresses $(x_1 \vee x_2 \vee x_3)$ in terms of "Protected OR", which corresponds to our relation M. From this, we express the other 3-clauses.

Lemma 8 *There exist a* $\mathrm{CNF}(S)$*-definable relation* Q' *and* $\mathbf{a}, \mathbf{b} \in Q'$ *such that the distance between them expands.*

Proof. Since S is not tight, it contains a relation Q which is not componentwise bijunctive. If Q contains $\mathbf{a}, \mathbf{b}$ where the distance between them expands, we are done. So assume that for all $\mathbf{a}, \mathbf{b} \in G(Q)$, $d_Q(\mathbf{a}, \mathbf{b}) = |\mathbf{a} - \mathbf{b}|$. Since Q is not componentwise bijunctive, there exists a triple of assignments $\mathbf{a}, \mathbf{b}, \mathbf{c}$ lying in the same component such that $\mathrm{maj}(\mathbf{a}, \mathbf{b}, \mathbf{c})$ is not in that component (which also easily implies it is not in Q). Choose the triple such that the sum of pairwise distances $d_Q(\mathbf{a}, \mathbf{b}) + d_Q(\mathbf{b}, \mathbf{c}) + d_Q(\mathbf{c}, \mathbf{a})$ is minimized. Let $U = \{i | a_i \neq b_i\}$, $V = \{i | b_i \neq c_i\}$, and $W = \{i | c_i \neq a_i\}$. Since $d_Q(\mathbf{a}, \mathbf{b}) = |\mathbf{a} - \mathbf{b}|$, a shortest path does not flip variables outside of U, and each variable in U is flipped exactly once. We note some useful properties of the sets U, V, W.

1) *Every index $i \in U \cup V \cup W$ occurs in exactly two of U, V, W.*
Consider going by a shortest path from $\mathbf{a}$ to $\mathbf{b}$ to $\mathbf{c}$ and back to $\mathbf{a}$. Every $i \in U \cup V \cup W$ is seen an even number of times along this path since we return to $\mathbf{a}$. It is seen at least once, and at most thrice, so in fact it occurs twice.
2) *Every pairwise intersection $U \cap V, V \cap W$ and $W \cap U$ is non-empty.*
Suppose the sets U and V are disjoint. From Property 1, we must have $W = U \cup V$. But then it is easy to see that $\mathrm{maj}(\mathbf{a}, \mathbf{b}, \mathbf{c}) = \mathbf{b}$ which is in Q. This contradicts the choice of $\mathbf{a}, \mathbf{b}, \mathbf{c}$.
3) *The sets $U \cap V$ and $U \cap W$ partition the set U.*
By Property 1, each index of U occurs in one of V and W as well. Also since no index occurs in all three sets U, V, W this is in fact a disjoint partition.
4) *For each index $i \in U \cap W$, it holds that $\mathbf{a} \oplus \mathbf{e}_i \notin Q$.*
Assume for the sake of contradiction that $\mathbf{a}' = \mathbf{a} \oplus \mathbf{e}_i \in R$. Since $i \in U \cap W$ we have simultaneously moved closer to both $\mathbf{b}$ and $\mathbf{c}$. Hence $d_Q(\mathbf{a}', \mathbf{b}) + d_Q(\mathbf{b}, \mathbf{c}) + d_Q(\mathbf{c}, \mathbf{a}') < d_Q(\mathbf{a}, \mathbf{b}) + d_Q(\mathbf{b}, \mathbf{c}) + d_Q(\mathbf{c}, \mathbf{a})$. Also $\mathrm{maj}(\mathbf{a}', \mathbf{b}, \mathbf{c}) = \mathrm{maj}(\mathbf{a}, \mathbf{b}, \mathbf{c}) \notin Q$. But this contradicts our choice of $\mathbf{a}, \mathbf{b}, \mathbf{c}$.

Property 4 implies that the shortest paths to $\mathbf{b}$ and $\mathbf{c}$ diverge at $\mathbf{a}$, since for any shortest path to $\mathbf{b}$ the first variable flipped is from $U \cap V$ whereas for a shortest path to $\mathbf{c}$ it is from $W \cap V$. Similar statements hold for the vertices $\mathbf{b}$ and $\mathbf{c}$. Thus along the shortest path from $\mathbf{a}$ to $\mathbf{b}$ the first bit flipped is from $U \cap V$ and the last bit flipped is from $U \cap W$. On the other hand, if we go from $\mathbf{a}$ to $\mathbf{c}$ and then to $\mathbf{b}$, all the bits from $U \cap W$ are flipped before the bits from $U \cap V$. We use this crucially to define Q'. We will add a set of 2-clauses that enforce the following rule on paths starting at $\mathbf{a}$: *Flip variables from $U \cap W$ before variables from $U \cap V$.* This will eliminate all shortest paths from $\mathbf{a}$ to $\mathbf{b}$ since they begin by flipping a variable in $U \cap V$ and end with $U \cap W$. The paths from $\mathbf{a}$ to $\mathbf{b}$ via $\mathbf{c}$ survive since they flip $U \cap W$ while going from $\mathbf{a}$ to $\mathbf{c}$ and $U \cap V$ while going from $\mathbf{c}$ to $\mathbf{b}$. However all remaining paths have length at least $|\mathbf{a} - \mathbf{b}| + 2$ since they flip twice some variables not in U.

Take all pairs of indices $\{(i, j) | i \in U \cap W, j \in U \cap V\}$. The following conditions hold from the definition of U, V, W: $a_i = \bar{c}_i = \bar{b}_i$ and $a_j = c_j = \bar{b}_j$. Add the 2-clause C_{ij} asserting that the pair of variables $x_i x_j$ must take values in $\{a_i a_j, c_i c_j, b_i b_j\} = \{a_i a_j, \bar{a}_i a_j, \bar{a}_i \bar{a}_j\}$. The new relation is $Q' = Q \wedge_{i,j} C_{ij}$. Note that $Q' \subset Q$. We verify that the distance between $\mathbf{a}$ and $\mathbf{b}$ in Q' expands. It is easy to see that for any $j \in U$, the assignment $\mathbf{a} \oplus \mathbf{e}_j \notin Q'$. Hence there are no shortest paths left from $\mathbf{a}$ to $\mathbf{b}$. On the other hand, it is easy to see that $\mathbf{a}$ and $\mathbf{b}$ are still connected, since the vertex $\mathbf{c}$ is still reachable from both. $\square$

Due to space constraints, all remaining proofs are in the full version.

5 Discussion and Open Problems

In Section 2, we conjectured a trichotomy for CONN(S). We have made progress towards this conjecture; what remains is to pinpoint the complexity of CONN(S) when S is Horn or dual-Horn. We can extend our dichotomy theorem for *st*-connectivity to formulas without constants; the complexity of connectivity for formulas without constants is open. We conjecture that when S is not tight, one can improve the diameter bound from $2^{\Omega(\sqrt{n})}$ to $2^{\Omega(n)}$. Finally, we believe that our techniques can shed light on other connectivity-related problems, such as approximating the diameter and counting the number of components.

References

1. Schaefer, T.: The complexity of satisfiability problems. In: Proc. 10^{th} ACM Symp. Theory of Computing. (1978) 216–226
2. Creignou, N.: A dichotomy theorem for maximum generalized satisfiability problems. Journal of Computer and System Sciences **51** (1995) 511–522
3. Creignou, N., Khanna, S., Sudan, M.: Complexity classification of Boolean constraint satisfaction problems. SIAM Monographs on Disc. Math. Appl. 7 (2001)
4. Khanna, S., Sudan, M., Trevisan, L., Williamson, D.: The approximability of constraint satisfaction problems. SIAM J. Comput., 30(6):1863–1920 (2001)
5. Creignou, N., Hermann, M.: Complexity of generalized satisfiability counting problems. Information and Computation **125**(1) (1996) 1–12
6. Kavvadias, D., Sideri, M.: The inverse satisfiability problem. SIAM J. Comput. **28**(1) (1998) 152–163
7. Kirousis, L., Kolaitis, P.: The complexity of minimal satisfiability problems. Information and Computation **187**(1) (2003) 20–39
8. Bulatov, A.: A dichotomy theorem for constraints on a three-element set. In: Proc. 43^{rd} IEEE Symp. Foundations of Computer Science. (2002) 649–658
9. Creignou, N., Zanuttini, B.: A complete classification of the complexity of propositional abduction. To appear in SIAM Journal on Computing (2006)
10. Achlioptas, D., Naor, A., Peres, Y.: Rigorous location of phase transitions in hard optimization problems. Nature **435** (2005) 759–764
11. Mézard, M., Zecchina, R.: Random k-satisfiability: from an analytic solution to an efficient algorithm. Phys. Rev. E **66** (2002)
12. Mézard, M., Parisi, G., Zecchina, R.: Analytic and algorithmic solution of random satisfiability problems. Science **297, 812** (2002)
13. Maneva, E., Mossel, E., Wainwright, M.J.: A new look at survey propagation and its generalizations. In: Proc. 16^{th} ACM-SIAM Symp. Discrete Algorithms. (2005) 1089–1098
14. Mora, T., Mézard, M., Zecchina, R.: Clustering of solutions in the random satisfiability problem. Phys. Rev. Lett. (2005) In press.
15. Achlioptas, D., Ricci-Tersenghi, F.: On the solution-space geometry of random constraint satisfaction problems. In: 38^{th} ACM Symp. Theory of Computing. (2006)
16. Selman, B., Kautz, H., Cohen, B.: Local search strategies for satisfiability testing. In: Cliques, coloring, and satisfiability : second DIMACS implementation challenge, October 1993, AMS (1996)
17. Achlioptas, D., Beame, P., Molloy, M.: Exponential bounds for DPLL below the satisfiability threshold. In: Proc. 15^{th} ACM-SIAM Symp. Discrete Algorithms. (2004) 132–133
18. Böhler, E., Creignou, N., Reith, S., Vollmer, H.: Playing with Boolean blocks, Part II: constraint satisfaction problems. ACM SIGACT-Newsletter **35**(1) (2004) 22–35
19. Hearne, R., Demaine, E.: The Nondeterministic Constraint Logic model of computation: Reductions and applications. In: 29^{th} Intl. Colloquium on Automata, Languages and Programming. (2002) 401–413