

Using extended resolution to represent strongly connected components of directed graphs

Gábor Kusper^{ab}, Zijian Gyöző Yang^d, Benedek Nagy^{ac}

^aEszterházy Károly Catholic University,
Faculty of Informatics,
Eger, Hungary

^bUniversity of Debrecen,
Faculty of Informatics,
Debrecen, Hungary

^cEastern Mediterranean University,
Faculty of Arts and Sciences,
Department of Mathematics,
Famagusta, North Cyprus, Mersin-10, Türkiye

^dHungarian Research Centre for Linguistics,
Budapest, Hungary

Abstract. In this paper, we study how to represent a directed graph as a SAT problem. We study those directed graphs which consists of two strongly connected components (SCC). We reuse the SAT models which are known as the Black-and-White SAT representations. We present the so-called 3rd Solution Lemma: If a directed graph consists of two SCCs, A and B , and there is an edge from A to B , then the corresponding SAT representation has 3 solutions: the black assignment, the white assignment, and the 3rd solution can be written as $\neg A$ union B . Using this result, we present an important negative result: We cannot represent all SAT problems as directed graphs using the Black-and-White SAT representations. Furthermore, we study the question how to represent an SCC by one Boolean variable to maintain the 3rd Solution Lemma. For that we use extended resolution.

Keywords: SAT problem, boolean logic, directed graphs

AMS Subject Classification: 03B05, 05C20

1. Introduction

There are various interesting links between directed graphs and SAT problems [11]. In this paper we are interested in such type of relationship. On the one hand, directed graphs could represent many types of objects, but in many cases, it is not so straightforward how and what type of representation leads to some advantages. Since, this problem generally seems to be very difficult, we work on a related one: we represent a directed graph as a propositional logical formula or, in fact, as a SAT problem.

We should mention here truth-teller–liar logical puzzles, when some people are having statements about the types of subsets of people and this information can be represented by graphs where the vertices representing the people. Depending on the various subtypes of the puzzles, the statements could produce a symmetric relation (strong truth-teller–strong liar, shortly SS puzzles [28]), meaning e.g., that the type of the persons are the same or the opposite. In many other puzzles, however, the statements are represented by directed edges and they represent implication type relations [27, 29, 30]. In some puzzles self-reference statements can also play important role [2, 3], and these puzzles are related also to the liar paradox [5]. The graph model of these puzzles usually used to infer some additional information and in many cases also the solution of the puzzle (i.e., assigning the types, like truth-teller and liar to each vertex in such a way that the statements match). Some puzzles are also analysed from the information flow point of view [6, 31].

There are also several other models to connect directed graphs to the SAT problem [20]. Each of these models has the following property: If the represented directed graph is strongly connected, then its SAT representation has only two solutions, the one where all variables are false, called the black assignment, and the one where all variables are true, called the white assignment. These models are called the Black-and-White SAT representations [9]. In this paper, we make a step further: we study those directed graphs which consist of not only one strongly connected component (SCC), but more. In this work we show that if a directed graph consists of two SCC components, A and B , and there is an edge from A to B , then the corresponding SAT representation has a third solution which is $\neg A$ union B . we call this result as the 3rd Solution Lemma.

This result a step forward in our main goal: Be able to represent any SAT problem as directed graph. Unfortunately, the final result is a negative one, we cannot achieve this goal based on Black-and-White SAT representations, as it is presented in Section 6.

We think that a graph representation conveys more intuitions than a logical formula. There are also neural networks which works on graphs. In the field of natural language processing, there are two main recent topics: large language models [32] and knowledge graphs [36]. Integration of knowledge graphs into language modeling became one of the most important research field [1, 35, 39]. Furthermore, Jiang, Gurajada et al. [16] takes an unorthodox view on the problem: combining textual heuristics together with neural features in a weighted rule-based framework

based on first-order logic. This research based on the Logical Neural Networks [34], which is also becoming an increasingly researched area [24, 25].

Furthermore, we study the question how to represent an SCC by one Boolean variable. We found out that extended resolution [12, 38] is a suitable tool for that. To represent the SCC which consists of only two vertices a and b , we have to add to its model the following formula: $a \wedge b$ equals x , i.e., the following clauses: $\neg a \vee \neg b \vee x$, $\neg x \vee a$, and $\neg x \vee b$. This is the classical example of extended resolution, where x is a new variable. Although, the original problem is still very difficult, this work helps us to understand better what extended resolution means, and how to represent extended resolution graphically.

2. Formal definitions

As usual in logic and in research about SAT [7, 8, 33] we define how our objects (i.e. formulae) and their representations build up.

A *literal* is a Boolean variable, called positive literal, or the negation of a Boolean variable, called negative literal. Examples for literals are: $a, \neg a, b, \neg b, \dots$

A *clause* is a set of literals. A *clause set* is a set of clauses. A *SAT problem* is a clause set. An *assignment* is a set of literals. In a clause or in an assignment, a variable may occur either as a positive literal or as a negative literal, but not as both, or it may not occur at all.

Clauses are interpreted as disjunction of their literals. Assignments are interpreted as conjunction of their literals. Clause sets are interpreted as conjunction of their clauses.

If a clause or an assignment contains exactly k literals, then we say it is a k -*clause* or a k -*assignment*, respectively. A 1-clause is called to be a *unit*, a 2-clause is called to be a *binary clause*. A k -*SAT problem* is a clause set where its clauses have at most k literals. A clause from a clause set is a *full-length clause* iff it contains all variables from the clause set.

We use two intuitive notions: *NNP clause*, and *NPP clause*. A clause is an *NNP clause* iff it contains exactly one positive literal. A clause is an *NPP clause* iff it contains exactly one negative literal.

Negation of a set H is denoted by $\neg H$ which means that all elements in H are negated. Note that $\neg\neg H = H$.

Let $Vars$ be the set of variables of a clause set. We say that WW is the *white clause* or the *white assignment* iff $WW = Vars$. We say that BB is the *black clause* or the *black assignment* iff $BB = \neg Vars$. For example if $Var = \{a, b, c\}$, then $WW = \{a, b, c\}$, and $BB = \{\neg a, \neg b, \neg c\}$.

We say that clause C *subsumes* clause D iff C is a subset of D .

We say that clause set S *subsumes* clause C iff there is a clause in S which subsumes C . Formally: S *subsumes* $C \iff \exists D(D \in S \wedge D \subseteq C)$.

We say that assignment M is a *solution* for clause set S iff for all $C \in S$ we have $M \cap C \neq \{ \}$.

We say that the clause set S is a *Black-and-White SAT problem* iff it has only two solutions, the white assignment (WW) and the black one (BB).

We say that clause sets A and B are equivalent, denoted by $A \equiv B$, iff A and B have the same set of solutions. We say that clause set A entails clause set B iff the set of solutions of A is a subset of the set of solutions of B , i.e., A may have no other solutions than B . This notion is denoted by $A \geq B$. Note that if A subsumes all clauses of B , then $A \geq B$.

We say that A is stronger than B iff $A \geq B$ and A and B are not equivalent. This notion is denoted by $A > B$.

Resolution $(\{a\} \cup A, \{-a\} \cup B) = A \cup B$, if $\{a\} \cup A$, $\{-a\} \cup B$, and $A \cup B$ are clauses. For example $\text{Resolution}(\{a, b\}, \{-a, c\}) = \{b, c\}$. But we cannot do resolution on $\{a, b\}$ and $\{-a, -b\}$, because $\{b, -b\}$ is not a clause. We say that literal c is blocked in clause C within clause set S iff for all $D \in S$ we have that if $\neg c \in D$ then $C \cup D \setminus \{c, \neg c\}$ is not a clause. In this case we say that C is blocked in S . It is well-known that a blocked clause can be deleted or added to a clause set without changing its satisfiability [17, 18] (but the set of solutions might be changed). If a is a literal in clause set S , and $\neg a$ is not a literal in S , then we say that a is a pure literal in S . Any pure literal is a blocked literal at the same time.

Extended resolution [12, 38] is SAT solver preprocessing technique to add blocked clauses to a clause set, so that the satisfiability of the original clause set is not changed. We use this technique to add (usually short) clauses which speed-up the search. Extended resolution adds clauses which are equal to $x \leftrightarrow f(\text{Var})$, where x is a new variable, and $f(\text{Var})$ is a Boolean function on some variables. The most widely used form is $x \leftrightarrow a \wedge b$, which is equivalent to the clause set: $\{\{-a, \neg b, x\}, \{a, \neg x\}, \{b, \neg x\}\}$. In this way x is blocked in all new clauses, so the new clauses are blocked.

Now, we recall some concepts of graph theory [4, 13].

The construction $D = (V, E)$ is a directed graph, where V is the set of vertices, and E is the set of edges. An edge is an ordered pair of vertices. The edge (a, b) is depicted by $a \rightarrow b$, and we can say that a has a child b . If (a, b) is an element of E , then we may say that (a, b) is an edge of D .

We say that $D = (V, E)$ is a communication graph iff for all a in V have that (a, a) is not in E , and if x is an element of V , then $\neg x$ must not be an element of V . We need this constraint because we generate a logical formula out of D . If we speak about a communication graph, then we may use the word node as a synonym of vertex.

A path from a_1 to a_j in directed graph D is a sequence of vertices a_1, a_2, \dots, a_j such that for each $i \in \{1, \dots, j-1\}$ we have that (a_i, a_{i+1}) is an edge of D . A path from a_1 to a_j in directed graph D is a *cycle* iff (a_j, a_1) is an edge of D . The cycle $a_1, a_2, \dots, a_j, a_1$ is represented by the following tuple: (a_1, a_2, \dots, a_j) . This tuple can be used as a set of its elements as we shall define it formally later. Note that in the representation of a cycle the first and the last element must not be the same vertex.

If we have a cycle (a_1, a_2, \dots, a_n) , then b is an exit point of it iff for some

$j \in \{1, 2, \dots, n\}$ we have that (a_j, b) is an edge and $b \notin \{a_1, a_2, \dots, a_n\}$.

A directed graph is complete iff every pair of distinct vertices is connected by a pair of unique edges (one in each direction). A directed graph is strongly connected iff there is a path from each vertex to each other vertex. Note that a complete graph is also strongly connected. Note that a strongly connected graph contains a cycle which contains all vertices.

The directed graph $G' = (V', E')$ is a subgraph of $G = (V, E)$ iff V' is a subset of V and E' is a subset of E . A subgraph of a directed graph G is a strongly connected component (SCC) iff it is strongly connected, and is maximal with this property: no additional edges or vertices from G can be included in the subgraph without breaking its property of being strongly connected. It is possible to test the strong connectivity of a graph, or to find its strongly connected components, in linear time, $O(|V| + |E|)$. The collection of strongly connected components forms a partition of the set of vertices of G .

We can create the so called condensation graph of a directed graph G by substituting each SCC of G by a single new vertex. The condensation graph is always a directed acyclic graph (DAG).

Somewhat similar technique was also used to solve truth-teller-liar puzzles by simplifying their graphs.

A SAT model (or shortly a model) of a communication graph is a mapping which maps nodes to boolean variables by a bijection, and which maps edges, cycles and possibly other parts of the graph to clauses, which creates a SAT problem, and from which the communication graph can be reconstructed up to those parts which are encoded. Generally, we use the function $MM(X)$ to assign a model to the communication graph X .

Especially, the Strong Model, our first model, was defined formally in our first paper [9]. First, we recall its definition.

Let D be a communication graph, then the Strong Model of D is denoted by $SM(D)$, and defined as follows:

$$SM(D) := \{\{\neg a, b\} \mid D = (V, E) \wedge (a, b) \in E\}.$$

Note that the the Strong Model can be the empty set. For example, if $D = (\{a, b\}, \{\})$, then $SM(D) = \{\}$.

The Strong Model of any communication graph must have the black solution (when every variable has the value true) and the white solution (when every variable has the value false), see Lemma 3.1.

In this way, actually, the Strong Model assigns an implication formula for each of the directed edges somewhat similarly as edges represent implications in some of the truth-teller-liar puzzles.

The Weak Model was defined formally in our second paper [21]. First, we recall its definition.

Let $D = (V, E)$ be a communication graph. Then we define the following notions:

$$OutE(a, E) := \{b \mid (a, b) \in E\}.$$

$$\text{NodeRep}(a, E) := \{\neg a\} \cup \text{Out}E(a, E).$$

$$\text{NodeRep}(D) := \{\text{NodeRep}(a, E) \mid D = (V, E) \wedge a \in V \wedge \text{Out}E(a, E) \neq \emptyset\}.$$

$$\text{Cycles}(D) := \{\{a_1, a_2, \dots, a_k\} \mid D = (V, E) \wedge k = 1 \vee$$

$$\forall_{i=1 \dots k} (a_{(i \bmod k)+1} \in \text{Out}E(a_i, E))\}.$$

$$\text{ExitPoints}(\{a_1, a_2, \dots, a_k\}, E) := \{b \mid \exists_{i=1 \dots k} (b \in \text{Out}E(a_i, E)) \wedge$$

$$\neg \exists_{j=1 \dots k} (b = a_j)\}.$$

$$\text{CycleRep}(D) := \{\neg C \cup \text{ExitPoints}(C, E) \mid D = (V, E) \wedge$$

$$C \in \text{Cycles}(D) \wedge \text{ExitPoints}(C, E) \neq \emptyset\}.$$

$$\text{WM}(D) := \text{NodeRep}(D) \cup \text{CycleRep}(D).$$

We say that $\text{WM}(D)$ is the Weak Model of D .

3. The Black-and-White SAT representations

We recall some important results from [21]. We do not recall the proofs, only the theorems. Then we define the notion of Black-and-White SAT representations, and give a new theorem which helps us to prove the 3rd Solution Lemma.

Lemma 3.1. *Let D be a communication graph. Then $\text{WM}(D)$ has at least two solutions, namely the white assignment (WW) and the black assignment (BB). The same is true for $\text{SM}(D)$.*

Theorem 3.2. *Let D be a communication graph. Then $\text{SM}(D)$ is a Black-and-White 2-SAT problem iff the graph D is strongly connected.*

Theorem 3.3. *Let D be a communication graph. Then $\text{WM}(D)$ is a Black-and-White SAT problem iff D is strongly connected.*

Theorem 3.4 (Transitions Theorem). *If $\text{MM}(X)$ is an arbitrary but fixed model of communication graphs, such that for any communication graph D we have that $\text{SM}(D) \geq \text{MM}(D) \geq \text{WM}(D)$, then $\text{MM}(D)$ is a Black-and-White SAT problem iff D is strongly connected.*

Until this point we have recalled some results from [21]. Now we give a new definition and a theorem. We shall define the notion of *Black-and-White SAT representations*, which is clearly motivated by the Transitions Theorem.

Definition 3.5. We say that function $\text{MM}(X)$, which generates a SAT problem from a communication graph, is a Black-and-White SAT representation iff for all communication graphs D we have that $\text{SM}(D) \geq \text{MM}(D) \geq \text{WM}(D)$.

There are some known example between the Strong and the Weak Model, for example: the Balatonboglár Model [21], and the Simplified Balatonboglár Model [22].

The Transitions Theorem can be extended easily for those cases where the two extreme cases, the Strong Model, and the Weak Model generate SAT problems with the same solution set. We give also the proof of this extended theorem because it is a new one.

Theorem 3.6. *Let $MM(X)$ be a Black-and-White SAT representation. Let D be an arbitrary but fixed communication graph. If $SM(D)$ and $WM(D)$ have the same set of solutions, then $SM(D)$, $MM(D)$, and $WM(D)$ have the same set of solutions.*

Proof. Let $MM(X)$ be a Black-and-White SAT representation. Let D be an arbitrary but fixed communication graph. From these we know that $SM(D) \geq MM(D) \geq WM(D)$. Assume that $SM(D)$ and $WM(D)$ have the same set of solutions. Then, by definition of clause set equivalence, we know that $SM(D) \equiv WM(D)$. From this and from $SM(D) \geq MM(D) \geq WM(D)$ we obtain that $SM(D) \equiv MM(D) \equiv WM(D)$. Hence, $SM(D)$, $MM(D)$, and $WM(D)$ have the same set of solutions. \square

The message of this theorem is the following. If the two extreme cases, the Strong Model, and the Weak Model share a property on the set of solutions, then any other model between them should also have that property. We are going to use this result to proof the 3rd Solution Lemma in the next section.

4. The 3rd Solution Lemma

In this section we give and prove the so called 3rd Solution Lemma. This lemma states the following. If we use a Black-and-White SAT representation to represent a communication graph, i.e., we have a property that an SCC with vertices A represented as a SAT problem has exactly two solutions: the black, and the white assignment, then a communication graph with exactly two SCCs, with vertex sets A and B , where we have some edges from A to B , has exactly 3 solutions: $A \cup B$, $\neg A \cup \neg B$, and $\neg A \cup B$.

This result a step forward in our main goal: Be able to represent any SAT problem as directed graph. Unfortunately, the final result is a negative one, we cannot achieve this goal based on Black-and-White SAT representations, as it is presented and proven in Section 6.

First we present an auxiliary lemma, where there is no edge between the two SCCs.

Lemma 4.1. *Let C be a communication graph which consists of exactly two separate SCCs: A , and B , such that there is no edge between them. Let S be a SAT model of C generated by a Black-and-White SAT representation. Then S has exactly 4 solutions: $A \cup B$ (which is the white assignment), $\neg A \cup \neg B$ (which is the black one), $\neg A \cup B$, and $A \cup \neg B$.*

Proof. We know that A and B are distinct sets of vertices, because otherwise they would not be separate SCCs.

First, let us assume that S has been generated by the Strong Model. Then, by Theorem 3.2, we know that the representation of A has two models A and $\neg A$, and the representation of B has two models B and $\neg B$. Since the set of vertices of C equals $A \cup B$ and there is no edge between A and B , and, furthermore, S is generated from C by the Strong Model, we obtain that solution of S are the combinations of the solutions of the representations of A and B , which are the following 4 ones: $A \cup B$, $A \cup \neg B$, $\neg A \cup B$, and $\neg A \cup \neg B$.

Secondly, let us assume that S is generated by the Weak Model. Then, by Theorem 3.3, we obtain that S has the same set of solutions using the same proof steps.

Now let us assume that S is generated by any other Black-and-White SAT representation. Then, from the previous two cases and from Theorem 3.6, we obtain that S has the same set of solutions.

Hence, if C is a communication graph which consists of exactly two separate SCCs: A , and B , such that there is no edge between them, and S is a SAT model of C generated by a Black-and-White SAT representation, then S has exactly 4 solutions: $A \cup B$ (which is the white assignment), $\neg A \cup \neg B$ (which is the black one), $\neg A \cup B$, and $A \cup \neg B$. \square

Now let us study the case that we have some edges between the two SCCs. We show that in this case the Black-and-White SAT representations results in 3 solution. This theorem will be called the 3rd Solution Lemma. This is our first main result.

Theorem 4.2 (3rd Solution Lemma). *Let C be a communication graph which consists of exactly two separate SCCs: A , and B , such that there is at least one edge from A to B . Let S be a SAT model of C generated by a Black-and-White SAT representation. Then S has exactly 3 solutions: $A \cup B$ (which is the white assignment), $\neg A \cup \neg B$ (which is the black one), and $\neg A \cup B$.*

Proof. We know from our auxiliary Lemma 4.1, that if there were no edges between A and B , then we would have 4 solutions: $A \cup B$, $A \cup \neg B$, $\neg A \cup B$, and $\neg A \cup \neg B$.

It is enough to show that solution $A \cup \neg B$ is excluded, and $\neg A \cup B$ is not excluded because of the edges from A to B .

We do not have to consider the solutions $A \cup B$, which is the white assignment, and $\neg A \cup \neg B$, which is the black assignment, because they are always solutions, see Lemma 3.1 and Theorem 3.4.

Let us assume that we have only one edge from A to B , the edge (a_i, b_j) , where $a_i \in A$, and $b_j \in B$. So (a_i, b_j) is an edge of C .

First, let us assume that S is generated from C by the Strong Model. Since (a_i, b_j) is an edge of C , we know, by definition of the Strong Model, that $\{\neg a_i, b_j\}$ is a clause in S . We know that $A \cup \neg B$ does not satisfy this clause, on the other hand $\neg A \cup B$ satisfies it. Therefore, S has 3 solutions: $A \cup B$, $\neg A \cup B$, and $\neg A \cup \neg B$.

If we have more than one edge from A to B , then the clauses generated from those edges have the same property, thus $\neg A \cup B$ satisfies them.

Secondly, let us assume that S is generated from C by the Weak Model. Again, let us start the discussion with the case of a sole edge from A to B . Since (a_i, b_j) is an edge of C , where a_i is part of the cycle A , and where b_j is the only one exit point of this cycle, because A , and B are two separate SCCs. From this we know, by definition of the Weak Model, that $\{\neg A, b_j\}$ is a clause in S . We know that $A \cup \neg B$ does not satisfy this clause, on the other hand $\neg A \cup B$ satisfies it. There might be other cycles which contains a_i , but those are subsets of A , and they have at least one exit point, b_j . Therefore, the clauses generated from those cycles are satisfied by $\neg A \cup B$. If we have more than one edge from A to B , then the cycles from A have more exit points to B , but still, those clauses generated from those cycles have the same property, thus $\neg A \cup B$ satisfies them.

Now let us assume that S is generated by any other Black-and-White SAT representation. Then, from the previous two cases and from Theorem 3.6, we obtain that S has the same set of solutions.

Hence, if C is a communication graph which consists of exactly two separate SCCs: A , and B , such that there is at least one edge from A to B , and S is a SAT model of C , then S has exactly 3 solutions: $A \cup B$ (which is the white assignment), $\neg A \cup \neg B$ (which is the black one), and $\neg A \cup B$. \square

Notice here that if the graph has two strongly connected components, then either there is no edge between them in any directions, or there is exactly one direction in which there is/are edge(s) between them in the graph. In this way, we have provided results for every graph that has exactly two SCCs.

5. An example with extended resolution

The 3rd Solution Lemma (Theorem 4.2) allows us to study not only strongly connected graphs, but also more general ones. In this section we give an example and some interesting connections to extended resolutions. We also raise some interesting questions which might highlight future ways of investigations.

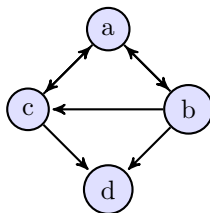


Figure 1. A communication graph with 4 vertices, 3 cycles.

Figure 1 shows a communication graph with 4 vertices, and 3 cycles.

As an example we show the Strong Model of the communication graph of Figure 1:

$$SM(D) = \{\{-a, b\}, \{-a, c\}, \{-b, a\}, \{-b, c\}, \\ \{-b, d\}, \{-c, a\}, \{-c, d\}\}. \quad (5.1)$$

Note that since the communication graph in Figure 1 is not strongly connected, its Strong Model (5.1) is not a Black-and-White SAT problem. Therefore, additionally to the black and white solutions, for example, it has the solution $\{-a, \neg b, \neg c, d\}$, as we expected because of the 3rd Solution Lemma (Theorem 4.2).

Note that, actually, this communication graph consists of two Strongly Connected Components (SCCs), which are: $\{a, b, c\}$ and $\{d\}$. Furthermore, there are edges from the first SCC to the second one, but not in the opposite direction. Hence, its model has to have, because of the 3rd Solution Lemma (Theorem 4.2), the solution: $\{-a, \neg b, \neg c, d\}$, which is the extra solution of our example.

Now, we can use extended resolution to introduce x instead of the first SCC. Thus, let x denote $a \wedge b \wedge c$. In this case we have to add the following clauses to the Strong Model of the graph: $\neg a \vee \neg b \vee \neg c \vee x$, $\neg x \vee a$, $\neg x \vee b$, and $\neg x \vee c$. It is easy to check that the new model still has only 3 solutions: $\{-a, \neg b, \neg c, \neg x, d\}$, $\{a, b, c, x, d\}$, and $\{-a, \neg b, \neg c, \neg x, \neg d\}$, the ones corresponding to the solutions of our original problem. This means that the first SCC can be substituted by the single variable x which greatly simplifies the graph and also its model.

Before we are going forward and show our other main result in the next section, we recall the following about another SAT representation technique: We know that resolvable networks can represent any SAT problem, see Lemma 3 in [23].

Now, some very interesting questions are arising. How to generalize the notion of SCC for resolvable networks? How can we recognize extended resolution? Especially, how can we recognize the new variable of extended resolution in a resolvable network?

Now, we switch back to the main topic of the present paper, the relation of communication graphs and SAT.

6. A negative result

Based on the 3rd Solution Lemma (Theorem 4.2), in this section, we concentrate on the question which SAT problems can be represented by communication graphs. Indeed, we can reverse engineering the directed graph representation of a SAT problem based on the solutions of the SAT problem.

We give all the examples with $N = 2$, and $N = 3$ variables. Note, that we do not list the black and the white assignments in set of solutions. (Technical note: We also used Wolfram Alpha to test our work.)

Examples with $N = 2$:

- Set of solutions: $\{\{-a, b\}\}$; the corresponding directed graph: $(\{a, b\}, \{(a, b)\})$.

- Set of solutions: $\{\{a, \neg b\}\}$; the corresponding directed graph: $(\{a, b\}, \{(b, a)\})$. Note that this is the same as the previous one up to variable renaming. So the resulting graphs are isomorphic.
- Set of solutions: $\{\{\neg a, b\}, \{a, \neg b\}\}$; the corresponding directed graph: $(\{a, b\}, \{\})$. Note that the set of edges is empty.

Examples with $N = 3$ (we do not list isomorphic examples):

- Set of solutions: $\{\{\neg a, \neg b, c\}\}$; the corresponding directed graph: $(\{a, b, c\}, \{(a, b), (b, a), (a, c)\})$. Note that we have two SCCs: $\{a, b\}$ and $\{c\}$, and an edge (a, c) between them. Further, we could also add the edge (b, c) , because it is all the same which edge do we have between two SCCs. See the 3rd Solution Lemma (Theorem 4.2).
- Set of solutions: $\{\{\neg a, b, c\}\}$; the corresponding directed graph: $(\{a, b, c\}, \{(a, b), (b, c), (c, b)\})$. Note that we have two SCCs: $\{a\}$ and $\{b, c\}$, and an edge (a, b) between them.
- Set of solutions: $\{\{\neg a, \neg b, c\}, \{\neg a, b, c\}\}$; the corresponding directed graph: $(\{a, b, c\}, \{(a, b), (b, c)\})$. Note that we have three SCCs: $\{a\}$, $\{b\}$, and $\{c\}$, and there is a single path which consist of the edges: $(a, b), (b, c)$.
- Set of solutions: $\{\{\neg a, \neg b, c\}, \{\neg a, b, \neg c\}, \{\neg a, b, c\}\}$; the corresponding directed graph: $(\{a, b, c\}, \{(a, b), (a, c)\})$. Note that we have three SCCs: $\{a\}$, $\{b\}$, and $\{c\}$, and they form a tree, where the root is a and the two leaves are b , and c .
- Set of solutions: $\{\{\neg a, \neg b, c\}, \{\neg a, b, \neg c\}\}$; the corresponding directed graph: Unfortunately, we cannot construct this. Based on this observation we have created Theorem 6.1, see below.

Now, we have arrived to the other of our main results. By listing some of the possibilities, it can clearly be seen that we have more possible SAT problems than how many communication graphs we have with the same number of variables (e.g., vertices). Thus, based on this counting argument, we can state our second main result:

Theorem 6.1. *There exists a SAT problem S for which there is no communication graph C such that $MM(C)$ is equivalent to S , if $MM(X)$ is a Black-and-White SAT representation.*

Proof. Let us consider all communication graph with 3 nodes. There are 16 possible ones up to isomorphism, see, e.g., https://mathinsight.org/image/three_node_motifs. We give the Strong and the Weak Model for all of them. Then we show that there is a SAT problem which cannot be represented, if we use a Black-and-White SAT representation.

1. $D_1 = (\{a, b, c\}, \{\}),$
 $SM(D_1) = \{\}, WM(D_1) = \{\},$ i.e.,
 $SM(D_1) \equiv WM(D_1).$ Number of SCCs is 3. All eight possible assignments are solutions, i.e., they satisfy the ‘empty’ condition.
2. $D_2 = (\{a, b, c\}, \{(a, b)\}),$
 $SM(D_2) = \{\{-a, b\}\}, WM(D_2) = \{\{-a, b\}\},$ i.e.,
 $SM(D_2) \equiv WM(D_2).$ Number of SCCs is 3.
3. $D_3 = (\{a, b, c\}, \{(a, b), (b, a)\}),$
 $SM(D_3) = \{\{-a, b\}, \{-b, a\}\}, WM(D_3) = \{\{-a, b\}, \{-b, a\}\},$ i.e.,
 $SM(D_3) \equiv WM(D_3).$ Number of SCCs is 2.
4. $D_4 = (\{a, b, c\}, \{(a, b), (a, c)\}),$
 $SM(D_4) = \{\{-a, b\}, \{-a, c\}\}, WM(D_4) = \{\{-a, b, c\}\},$ i.e.,
 $SM(D_4) \geq WM(D_4),$
 $SM(D_4) \equiv WM(D_4) \cup \{\{-a, b, \neg c\}, \{-a, \neg b, c\}\}.$ Number of SCCs is 3.
5. $D_5 = (\{a, b, c\}, \{(b, a), (c, a)\}),$
 $SM(D_5) = \{\{-b, a\}, \{-c, a\}\}, WM(D_5) = \{\{-b, a\}, \{-c, a\}\},$ i.e.,
 $SM(D_5) \equiv WM(D_5).$ Number of SCCs is 3.
6. $D_6 = (\{a, b, c\}, \{(c, a), (a, b)\}),$
 $SM(D_6) = \{\{-c, a\}, \{-a, b\}\}, WM(D_6) = \{\{-c, a\}, \{-a, b\}\},$ i.e.,
 $SM(D_6) \equiv WM(D_6).$ Number of SCCs is 3.
7. $D_7 = (\{a, b, c\}, \{(c, a), (a, b), (b, a)\}),$
 $SM(D_7) = \{\{-c, a\}, \{-a, b\}, \{-b, a\}\},$
 $WM(D_7) = \{\{-c, a\}, \{-a, b\}, \{-b, a\}\},$ i.e.,
 $SM(D_7) \equiv WM(D_7).$ Number of SCCs is 2.
8. $D_8 = (\{a, b, c\}, \{(a, c), (a, b), (b, a)\}),$
 $SM(D_8) = \{\{-a, c\}, \{-a, b\}, \{-b, a\}\},$
 $WM(D_8) = \{\{-a, b, c\}, \{-b, a\}, \{-a, \neg b, c\}\},$ i.e.,
 $SM(D_8) \geq WM(D_8), SM(D_8) \equiv WM(D_8) \cup \{\{-a, b, \neg c\}\}.$ Number of SCCs is 2.
9. $D_9 = (\{a, b, c\}, \{(a, c), (c, a), (a, b), (b, a)\}),$
 $SM(D_9) = \{\{-a, c\}, \{-c, a\}, \{-a, b\}, \{-b, a\}\},$
 $WM(D_9) = \{\{-a, b, c\}, \{-c, a\}, \{-b, a\}, \{-a, \neg b, c\}, \{-a, \neg c, b\}\},$ i.e.,
 $SM(D_9) \equiv WM(D_9),$ because of resolution. Number of SCCs is 1.

10. $D_{10} = (\{a, b, c\}, \{(a, b), (a, c), (c, b)\})$,
 $SM(D_{10}) = \{\{-a, b\}, \{-a, c\}, \{-c, b\}\}$, $WM(D_{10}) = \{\{-a, b, c\}, \{-c, b\}\}$,
i.e.,
 $SM(D_{10}) \geq WM(D_{10})$, $SM(D_{10}) \equiv WM(D_{10}) \cup \{\{-a, \neg b, c\}\}$. Number of SCCs is 3.
11. $D_{11} = (\{a, b, c\}, \{(a, b), (b, c), (c, a)\})$,
 $SM(D_{11}) = \{\{-a, b\}, \{-b, c\}, \{-c, a\}\}$,
 $WM(D_{11}) = \{\{-a, b\}, \{-b, c\}, \{-c, a\}\}$, i.e.,
 $SM(D_{11}) \equiv WM(D_{11})$. Number of SCCs is 1.
12. $D_{12} = (\{a, b, c\}, \{(a, b), (b, a), (c, a), (c, b)\})$,
 $SM(D_{12}) = \{\{-a, b\}, \{-b, a\}, \{-c, a\}, \{-c, b\}\}$,
 $WM(D_{12}) = \{\{-a, b\}, \{-b, a\}, \{-c, a, b\}\}$, i.e.,
 $SM(D_{12}) \equiv WM(D_{12})$, because of resolution. Number of SCCs is 2.
13. $D_{13} = (\{a, b, c\}, \{(a, b), (b, a), (a, c), (c, b)\})$,
 $SM(D_{13}) = \{\{-a, b\}, \{-b, a\}, \{-a, c\}, \{-c, b\}\}$,
 $WM(D_{13}) = \{\{-a, b, c\}, \{-b, a\}, \{-c, b\}, \{-a, \neg b, c\}\}$, i.e.,
 $SM(D_{13}) \equiv WM(D_{13})$, because of resolution. Number of SCCs is 1.
14. $D_{14} = (\{a, b, c\}, \{(a, b), (b, a), (a, c), (b, c)\})$,
 $SM(D_{14}) = \{\{-a, b\}, \{-b, a\}, \{-a, c\}, \{-b, c\}\}$,
 $WM(D_{14}) = \{\{-a, b, c\}, \{-b, a, c\}, \{-a, \neg b, c\}\}$, i.e.,
 $SM(D_{14}) \geq WM(D_{14})$, $SM(D_{14}) \equiv WM(D_{14}) \cup \{\{-a, b, \neg c\}, \{-b, \neg c, a\}\}$.
Number of SCCs is 2.
15. $D_{15} = (\{a, b, c\}, \{(a, b), (b, a), (a, c), (c, a), (c, b)\})$,
 $SM(D_{15}) = \{\{-a, b\}, \{-b, a\}, \{-a, c\}, \{-c, a\}, \{-c, b\}\}$,
 $WM(D_{15}) = \{\{-a, b, c\}, \{-b, a\}, \{-c, a, b\}, \{-a, \neg b, c\}, \{-a, \neg c, b\}\}$, i.e.,
 $SM(D_{15}) \equiv WM(D_{15})$, because of resolution. Number of SCCs is 1.
16. $D_{16} = (\{a, b, c\}, \{(a, b), (b, a), (a, c), (c, a), (b, c), (c, b)\})$,
 $SM(D_{16}) = \{\{-a, b\}, \{-b, a\}, \{-a, c\}, \{-c, a\}, \{-b, c\}, \{-c, b\}\}$, $WM(D_{16}) =$
 $\{\{-a, b, c\}, \{-b, a, c\}, \{-c, a, b\}, \{-a, \neg b, c\}, \{-a, \neg c, b\}, \{-b, \neg c, a\}\}$,
i.e.,
 $SM(D_{16}) \equiv WM(D_{16})$, because of resolution. Number of SCCs is 1.

Now, let us consider the special case when $S = \{\{\neg a, b\}, \{\neg a, c\}, \{a, \neg b, \neg c\}\}$. The solution set of S is $\{\{\neg a, \neg b, c\}, \{\neg a, b, \neg c\}\} \cup \{BB, WW\}$.

If we check, then neither of the above communication graph makes this property true: $SM(D_i) \geq S \geq WM(D_i)$, $i = 1 \dots 16$. To be more formal, we have to check all 16 cases. We do this in the following few paragraphs.

It is clear that there is no equivalent 2-SAT problem to S because all literals in its last clause are blocked. So we do not need to check those communication graphs where $SM(D_i) \equiv WM(D_i)$, since the Strong Model is always a 2-SAT problem. So we need to check only the following cases: $i = \{4, 8, 10, 14\}$.

By definition of clause set equality it is enough to show that either the solution set of $SM(D_i)$ is not a subset of solution set of S , or the solution set of S is not a subset of the solution set of $WM(D_i)$, where $i = \{4, 8, 10, 14\}$. We show all the 4 cases, $i = \{4, 8, 10, 14\}$, that this is true. Since the black and the white assignments are always solutions, we do not show them in the solution sets.

In the following 4 lines the first set is the set of solutions of $SM(D_i)$. the second one is the solution set of S , and the last one is the solution set of $WM(D_i)$.

$$\begin{aligned}
 i = 4: & \{\{\neg a, \neg b, c\}, \{\neg a, b, \neg c\}, \{\neg a, b, c\}\} \not\subseteq \{\{\neg a, \neg b, c\}, \{\neg a, b, \neg c\}\} \\
 \subseteq & \{\{\neg a, \neg b, c\}, \{\neg a, b, \neg c\}, \{\neg a, b, c\}, \{\neg a, b, \neg c\}, \{a, \neg b, c\}, \{a, b, \neg c\}\}. \\
 i = 8: & \{\{\neg a, \neg b, c\}\} \subseteq \{\{\neg a, \neg b, c\}, \{\neg a, b, \neg c\}\} \not\subseteq \{\{\neg a, \neg b, c\}, \{a, \neg b, c\}\}. \\
 i = 10: & \{\{\neg a, b, \neg c\}, \{\neg a, b, c\}\} \not\subseteq \{\{\neg a, \neg b, c\}, \{\neg a, b, \neg c\}\} \\
 \not\subseteq & \{\{\neg a, b, \neg c\}, \{\neg a, b, c\}, \{a, b, \neg c\}\}. \\
 i = 14: & \{\{\neg a, \neg b, c\}\} \subseteq \{\{\neg a, \neg b, c\}, \{\neg a, b, \neg c\}\} \\
 \not\subseteq & \{\{\neg a, \neg b, c\}, \{\neg a, b, c\}, \{a, \neg b, c\}\}.
 \end{aligned}$$

Therefore, for S there exists no communication graph D_i , $i = 1 \dots 16$ such that $MM(D_i)$ is equivalent to S , if $MM(X)$ is a Black-and-White SAT representation.

There are two isomorphic clause set for S : $S' = \{\{a, \neg b\}, \{\neg b, c\}, \{\neg a, b, \neg c\}\}$; and $S'' = \{\{b, \neg c\}, \{a, \neg c\}, \{\neg a, \neg b, c\}\}$.

By similar steps we can show that there exists no communication graph D_i , $i = 1 \dots 16$ such that $MM(D_i)$ is equivalent to S' or S'' , if $MM(X)$ is a Black-and-White SAT representation.

Hence, there exists a SAT problem S for which there exists is no communication graph C such that $MM(C)$ is equivalent to S , if $MM(X)$ is a Black-and-White SAT representation. \square

The above proof is rather technical, but we decided to present it, because it was not easy to construct it.

We present a more intuitive proving argument as a second proof of Theorem 6.1.

Proof. Let us count first the number of different communication graphs (up to isomorphism). In case of 3 vertices there are 16 different communication graphs (up to isomorphism), see https://mathinsight.org/image/three_node_motifs. Actually, the number of strongly connected graphs with up to isomorphism is known for many decades ago [26] and shown in various textbooks, e.g., [10, 14, 15]. Various combinatorial relations and the integer sequence of the number of directed graphs as a function of the number of vertices is listed in [37] at <https://oeis.org/A000273>.

Now, let us consider the 3-variable SAT problems. They may have at least the following 17 different solution sets (up to isomorphism). The black and white solutions, i.e., ppp and nnn are not listed, but should be added to each set below.

There is a case, when no extra solution exists (the Black and White case).

The cases with a sole extra solution:

npp pnn .

With two extra solutions:

npp, pnp npp, pnn npp, npn pnn, npn .

With three extra solutions:

npp, pnp, ppn npp, pnp, pnn pnn, npn, npp pnn, npn, nnp
 npp, pnp, nnp pnn, npn, ppn .

With four extra solutions:

npp, pnp, ppn, pnn npp, pnp, pnn, npn npp, pnp, pnn, nnp pnn, npn, nnp, npp .

(And we have some cases with 5 or 6 extra solutions as well...)

Thus the number of possible solution sets is larger than the number of communication graphs and thus, the proof of the result is finished. \square

We decided to keep both proofs, because their structure are very different, and we think that both of them can be interesting for the readers. These two alternative proofs suggest that if we would like save our original goal, i.e., to find a graph representation of SAT problems, then we have to decrease the number of possible SAT instances, or we have to weaken the Weak Model or find some other representation.

We can decrease the number of possible SAT instances if we delete the blocked clauses before creating the corresponding directed graph. It is not so difficult to detect all blocked literal in case of 3-SAT, see [19], but in general, we have no idea how to show that this direction could work.

We can also use other representations. Actually, we have another candidate, the so called resolvable networks [23]. Its idea is that each clause can be represented by an $A \rightarrow B$ edge, where A represents the negative literals of the clause, and B represents the positive literals of it. A and B are called subnetworks, and they might have their own inner structure.

7. Conclusions

We have proven the 3rd Solution Lemma by showing that communication graphs with exactly two SSCs must have a 3rd solution in case of Black-and-White SAT representations. They must have 4 solution if there is no edges between the two SSCs, and they have 3 ones, if there are some edges (but not in both directions). We have also used a counting argument to prove that communications graphs cannot represent all possible SAT problems (if each variable is represented by a vertex). One may feel that this negative result gives a large imitation of our study, but this is actually, not the case. On the one hand, we believe that it gives a new motivation also to find and work with other type of graph representations of SAT, and not

only with pure communication graphs. On the other hand, the subclass of SAT problems that can be represented by communication graphs can still be interesting to find out and graphically, visually show how some techniques could help us to solve some of the SAT instances.

Acknowledgements. The authors would like to thank to the project “Quality insurance automatization services for IT development by analysing and refactoring of decision structures of source codes based on expert systems”, project ID: 2018-1.1.1-MKI-2018-00200, to support this research.

References

- [1] M. ALI, M. BERRENDORF, C. T. HOYT, L. VERMUE, S. SHARIFZADEH, V. TRESP, J. LEHMANN: *PyKEEN 1.0: A Python Library for Training and Evaluating Knowledge Graph Embeddings*, Journal of Machine Learning Research 22.82 (2021), pp. 1–6, URL: <http://jmlr.org/papers/v22/20-825.html>.
- [2] L. ALZBOON, B. NAGY: *Crazy Truth-Teller-Liar Puzzles*, Axiomathes 32.4 (2022), pp. 639–657.
- [3] L. ALZBOON, B. NAGY: *Truth-Teller-Liar Puzzles with Self-Reference*, Mathematics 8.2 (2020), paper 190, DOI: [10.3390/math8020190](https://doi.org/10.3390/math8020190).
- [4] J. BANG-JENSEN, G. GUTIN: *Digraphs: Theory, Algorithms and Applications*, Springer, Berlin, Heidelberg, Budapest, 2007.
- [5] J. BARWISE, L. S. MOSS: *Vicious Circles: On the Mathematics of Non-Wellfounded Phenomena*, CLSI Publications, 1996.
- [6] J. BARWISE, J. SELIGMAN: *Information Flow: the Logic of Distributed Systems*, Cambridge University Press, 1997.
- [7] M. BEN-ARI: *Mathematical Logic for Computer Science*, Springer, London, 2012.
- [8] A. BIÈRE, M. HEULE, H. VAN MAAREN, T. WALSH, eds.: *Handbook of Satisfiability*, IOS Press, 2021.
- [9] C. BIRÓ, G. KUSPER: *Equivalence of Strongly Connected Graphs and Black-and-White 2-SAT Problems*, Miskolc Mathematical Notes 19.2 (2018), pp. 755–768, DOI: [10.18514/MMN.2018.2140](https://doi.org/10.18514/MMN.2018.2140).
- [10] C. J. COLBOURN, J. H. DINITZ, eds.: *CRC Handbook of Combinatorial Designs*, CRC Press, 1996.
- [11] S. A. COOK: *The complexity of theorem proving procedures*, in: Proceedings of the Third Annual ACM Symposium, ACM, 1971, pp. 151–158.
- [12] S. A. COOK: *A Short Proof of the Pigeon Hole Principle Using Extended Resolution*, SIGACT News 8.4 (1976), pp. 28–32, ISSN: 0163-5700, DOI: [10.1145/1008335.1008338](https://doi.org/10.1145/1008335.1008338).
- [13] R. DIESTEL: *Graph Theory*, Springer, 2017.
- [14] J. L. GROSS, J. YELLEN, eds.: *Handbook of Graph Theory*, CRC Press, 2004.
- [15] F. HARARY, E. M. PALMER: *Graphical Enumeration*, Academic Press, NY, 1973.
- [16] H. JIANG, S. GURAJADA, Q. LU, S. NEELAM, L. POPA, P. SEN, Y. LI, A. GRAY: *LNN-EL: A Neuro-Symbolic Approach to Short-text Entity Linking*, in: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Online: Association for Computational Linguistics, Aug. 2021, pp. 775–787, DOI: [10.18653/v1/2021.acl-long.64](https://doi.org/10.18653/v1/2021.acl-long.64), URL: <https://aclanthology.org/2021.acl-long.64>.

- [17] O. KULLMANN: *New methods for 3-SAT decision and worst-case analysis*, Theoretical Computer Science 223.1-2 (1999), pp. 1–72.
- [18] O. KULLMANN: *On a Generalization of Extended Resolution*, Discrete Applied Mathematics 96-97.1-3 (1999), pp. 149–176.
- [19] G. KUSPER: *Finding Models for Blocked 3-SAT Problems in Linear Time by Systematical Refinement of a Sub-model*, in: KI 2006: Advances in Artificial Intelligence, Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 128–142, ISBN: 978-3-540-69912-5.
- [20] G. KUSPER, T. BALLA, C. BIRÓ, T. TAJTI, Z. G. YANG, I. BAJÁK: *Generating Minimal Unsatisfiable SAT Instances from Strong Digraphs*, in: 2020 22nd International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC), 2020, pp. 84–92, DOI: [10.1109/SYNASC51798.2020.00024](https://doi.org/10.1109/SYNASC51798.2020.00024).
- [21] G. KUSPER, C. BIRÓ: *Convert a Strongly Connected Directed Graph to a Black-and-White 3-SAT Problem by the Balatonboglár Model*, Algorithms 13.12 (2020), ISSN: 1999-4893, DOI: [10.3390/a13120321](https://doi.org/10.3390/a13120321), URL: <https://www.mdpi.com/1999-4893/13/12/321>.
- [22] G. KUSPER, C. BIRÓ, T. BALLA: *Representing Directed Graphs as 3-SAT Problems using the Simplified Balatonboglár Model*, in: The 11th International Conference on Applied Informatics (ICAI-2020), 2020, poster presented at ICAI-2020.
- [23] G. KUSPER, C. BIRÓ, B. NAGY: *Resolvable Networks—A Graphical Tool for Representing and Solving SAT*, Mathematics 9.20 (2021), ISSN: 2227-7390, DOI: [10.3390/math9202597](https://doi.org/10.3390/math9202597), URL: <https://www.mdpi.com/2227-7390/9/20/2597>.
- [24] T. LEBESE, N. MAKONDO, C. CORNELIO, N. KHAN: *Proof Extraction for Logical Neural Networks*, in: Advances in Programming Languages and Neurosymbolic Systems workshop at NeurIPS 2021, 2021.
- [25] S. LU, N. KHAN, I. Y. AKHALWAYA, R. RIEGEL, L. HOESH, A. GRAY: *Training Logical Neural Networks by Primal–Dual Methods for Neuro-Symbolic Reasoning*, in: ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2021, pp. 5559–5563, DOI: [10.1109/ICASSP39728.2021.9415044](https://doi.org/10.1109/ICASSP39728.2021.9415044).
- [26] M. D. MCILROY: *Calculation of numbers of structures of relations on finite sets*, Massachusetts Institute of Technology, Research Laboratory of Electronics, Quarterly Progress Reports, No. 17 (1955), pp. 14–22.
- [27] B. NAGY: *Duality of logical puzzles of type SW and WS—their solution using graphs*, Pure Math. Appl. 15.2-3 (2004), pp. 235–252.
- [28] B. NAGY: *SS-type truth-teller-liar puzzles and their graphs. (Hungarian)*, Alkalmaz. Mat. Lapok 23.1 (2006), pp. 59–72.
- [29] B. NAGY: *SW-type puzzles and their graphs*, Acta Cybern. 16.1 (2003), pp. 67–82, URL: <https://cyber.bibl.u-szeged.hu/index.php/actcybern/article/view/3611>.
- [30] B. NAGY: *Truth-teller and liar puzzles and their graphs*, CEJOR: Cent. Eur. J. Oper. Res. 11.1 (2003), pp. 57–72.
- [31] B. NAGY, G. ALLWEIN: *Diagrams and Non-monotonicity in Puzzles*, in: Diagrammatic Representation and Inference, Third International Conference, Diagrams 2004, Cambridge, UK, March 22-24, 2004, Proceedings, ed. by A. F. BLACKWELL, K. MARRIOTT, A. SHIMOJIMA, vol. 2980, Lecture Notes in Computer Science, Springer, 2004, pp. 82–96, DOI: [10.1007/978-3-540-25931-2_10](https://doi.org/10.1007/978-3-540-25931-2_10).
- [32] L. OUYANG, J. WU, X. JIANG, D. ALMEIDA, C. WAINWRIGHT, P. MISHKIN, C. ZHANG, S. AGARWAL, K. SLAMA, A. GRAY, J. SCHULMAN, J. HILTON, F. KELTON, L. MILLER, M. SIMENS, A. ASKELL, P. WELINDER, P. CHRISTIANO, J. LEIKE, R. LOWE: *Training language models to follow instructions with human feedback*, in: Advances in Neural Information Processing Systems, ed. by A. H. OH, A. AGARWAL, D. BELGRAVE, K. CHO, 2022.
- [33] K. PÁSZTORNÉ-VARGA, M. VÁRTERÉSZ: *A matematikai logika alkalmazásszemléletű tárgyalása (Application-oriented Study of Mathematical Logics, Hungarian)*, PANEM, Budapest, 2003.

- [34] R. RIEGEL, A. G. GRAY, F. P. S. LUUS, N. KHAN, N. MAKONDO, I. Y. AKHALWAYA, H. QIAN, R. FAGIN, F. BARAHONA, U. SHARMA, S. IKBAL, H. KARANAM, S. NEELAM, A. LIKHYANI, S. K. SRIVASTAVA: *Logical Neural Networks*, CoRR abs/2006.13155 (2020), arXiv: [2006.13155](https://arxiv.org/abs/2006.13155).
- [35] A. SAXENA, A. KOCHSIEK, R. GEMULLA: *Sequence-to-Sequence Knowledge Graph Completion and Question Answering*, in: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Dublin, Ireland: Association for Computational Linguistics, May 2022, pp. 2814–2828, DOI: [10.18653/v1/2022.acl-long.201](https://doi.org/10.18653/v1/2022.acl-long.201), URL: <https://aclanthology.org/2022.acl-long.201>.
- [36] P. SCHNEIDER, T. SCHOPF, J. VLADIKI, M. GALKIN, E. SIMPERL, F. MATTHES: *A Decade of Knowledge Graphs in Natural Language Processing: A Survey*, in: Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Online only: Association for Computational Linguistics, Nov. 2022, pp. 601–614, URL: <https://aclanthology.org/2022.aacl-main.46>.
- [37] N. J. A. SLOANE, S. PLOUFFE: *The Encyclopedia of Integer Sequences (and its online version on the link)*, Academic Press, 1995, URL: <https://oeis.org/>.
- [38] G. S. TSEITIN: *On the complexity of derivation in propositional calculus*, Structures in Constructive Mathematics and Mathematical Logic (1968), pp. 115–125.
- [39] R. WANG, D. TANG, N. DUAN, Z. WEI, X. HUANG, J. JI, G. CAO, D. JIANG, M. ZHOU: *K-Adapter: Infusing Knowledge into Pre-Trained Models with Adapters*, in: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, Online: Association for Computational Linguistics, Aug. 2021, pp. 1405–1418, DOI: [10.18653/v1/2021.findings-acl.121](https://doi.org/10.18653/v1/2021.findings-acl.121), URL: <https://aclanthology.org/2021.findings-acl.121>.