Graph-theoretic formulation of 2SAT

Denis TRYSTRAM

november 2020

Preliminaries

Satisfiability problems deal with propositional formulae that are populated by entities that can assume the truth-values TRUE and FALSE.

The entities are *logical variables*.

The *actual* entities that appear in each formula are *logical literals* (instances of logical variables) in either their *true* or *complemented* forms.

- In its *true* form, a literal evaluates to TRUE precisely when its associated variable does.
- In its *complemented* form, a literal evaluates to TRUE precisely when its associated variable evaluates to FALSE.

The following expression exemplify the notions.

formula:	$\Phi = (\neg x \vee y) \wedge (x \vee \neg y)$
variables:	x and y
literals:	x and y (true form); $\neg x$ and $\neg y$ (complemented form)

$$\begin{array}{ccccccccccc} (& \neg x & \vee & y &) & \wedge & (& x & \vee & \neg y &) \\ & \uparrow & & \uparrow & & & & \uparrow & & \uparrow & \\ & \text{complemented} & & \text{true} & & & & \text{true} & & \text{complemented} & \\ & \text{literal} & & \text{literal} & & & & \text{literal} & & \text{literal} & \end{array}$$

Recall the SAT problem

The Satisfiability Problem is specified by a propositional formula Φ that is a *conjunction of disjuncts of logical literals*.

The Satisfiability question is:
Can one assign truth-values to all of the logical variables of formula Φ in such a way that every disjunct evaluates to TRUE?

$$\Phi = C_1 \ \wedge \ C_2 \ \wedge \cdots \wedge \ C_m$$

where each clause $C_i \ = \ \ell_{i,1} \vee \ell_{i,2}$

Recall the SAT problem

The Satisfiability Problem is specified by a propositional formula Φ that is a *conjunction of disjuncts of logical literals*.

The Satisfiability question is:
Can one assign truth-values to all of the logical variables of formula Φ in such a way that every disjunct evaluates to $\textsc{true}$?

$$\Phi = C_1 \;\wedge\; C_2 \;\wedge \cdots \wedge\; C_m$$

where each clause $C_i \;=\; \ell_{i,1} \vee \ell_{i,2}$

2SAT is the particular problem where the clauses are of cardinality 2, example:
$\Phi_1 \;=\; (a \vee \neg b) \;\wedge\; (b \vee \neg c) \;\wedge\; (c \vee \neg a)$

Construction of the graph related to 2SAT

We transform Φ into a directed graph $G(\Phi)$ that has $2n$ vertices and $2m$ arcs.

- For each logical variable x there is one vertex that represents the TRUE literal form of variable x, and a second vertex that represents the FALSE literal form, $\neg x$, of the variable.
- Each clause $C_i = (\ell_{i,1} \vee \ell_{i,2})$ is represented by a pair of arcs.
  - There is an arc $(\neg x_1 \rightarrow x_2)$, which indicates that if x_1 is assigned truth-value FALSE, then x_2 should be assigned TRUE.
  - Symmetrically, there is an arc $(\neg x_2 \rightarrow x_1)$.

All paths in $G(\Phi)$ represent logical implications:

Representation of the graph for Φ_1

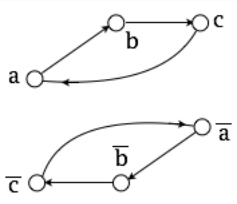
$$\Phi_1 \;=\; (a \vee \neg b) \;\wedge\; (b \vee \neg c) \;\wedge\; (c \vee \neg a)$$

In the first clause, if a is assigned FALSE then $\neg b$ is assigned TRUE (and thus, b is assigned FALSE).
Same for b and c in the two next clauses.

Representation of the graph for Φ_1

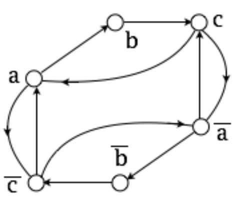
$$\Phi_1 \ = \ (a \vee \neg b) \ \wedge \ (b \vee \neg c) \ \wedge \ (c \vee \neg a)$$

In the first clause, if a is assigned FALSE then $\neg b$ is assigned TRUE (and thus, b is assigned FALSE).
Same for b and c in the two next clauses.



Another example for Φ_2

$$\Phi_2 \;=\; (a \vee \neg b) \;\wedge\; (b \vee \neg c) \;\wedge\; (c \vee \neg a) \;\wedge\; (a \vee c) \;\wedge\; (\neg a \vee \neg c)$$



Solving the problem

The idea here is to solve 2SAT by a path problem in graph $G(\Phi)$.

We prove first two technical lemmas.

Lemmas.

1. If $G(\Phi)$ contains a path from vertex x to vertex y, then it contains a path from vertex $\neg y$ to vertex $\neg x$.
2. If $G(\Phi)$ contains a path from vertex x to vertex y, then for every truth assignment τ that satisfies Φ, if τ assigns variable x the truth-value TRUE, then τ also assigns variable y the truth-value TRUE.

Proof of Lemma 1

The proof is simple because of the commutativity of the boolean operator $\vee$

Proof of Lemma 2

Assume that Φ is satisfied by a truth assignment τ which assigns variable x the truth-value TRUE.

- Say, for contradiction, that along the path from x to $\neg x$ in $G(\Phi)$, there exists an arc $(u \rightarrow v)$ such that assignment τ assigns TRUE to u and FALSE to v.
- Because of the way we constructed $G(\Phi)$, the existence of this arc means that Φ contains the clause $(\neg u \vee v)$.
- Moreover, under truth assignment τ, this clause of Φ evaluates to FALSE because both of its literals are assigned FALSE.
- This contradicts the assumption that τ is a satisfying assignment for Φ.

Solving the problem: the central theorem

Theorem.
The POS formula $\Phi = C_1 \wedge C_2 \wedge \cdots \wedge C_m$ is satisfiable if, and only if, no strongly connected component of $G(\Phi)$ contains both the positive form (x) and the negated form $(\neg x)$ of any variable x of Φ.

The theorem is an equivalence, thus, both sides (necessity and sufficiency) must be proved.

Principle of the proof

The overall form of the assertion we wish to prove is on the form

$$\mathsf{P} \;\Leftrightarrow\; \mathsf{Q}$$

This is a shorthand for the conjunction

$$\big[\mathsf{P} \;\Rightarrow\; \mathsf{Q}\big] \;\; \textbf{and} \;\; \big[\mathsf{Q} \;\Rightarrow\; \mathsf{P}\big]$$

We are able to derive a simplified proof here by replacing one of the two implications by its *contrapositive*; i.e., instead of proving the implication

$$\mathsf{P} \;\Rightarrow\; \mathsf{Q}$$

for the *necessity* component of the equivalence, we prove the *logically equivalent* implication

$$\neg\mathsf{Q} \;\Rightarrow\; \neg\mathsf{P}$$

Proof

Necessity

Say first that $G(\Phi)$ has a strongly connected component which contains vertices arising from a variable x in both positive (x) and negated ($\neg x$) forms.
We claim that Φ is not satisfiable.

By definition of "strongly connected", $G(\Phi)$ must contain paths between the vertices corresponding to x and to $\neg x$.
By Lemma 2, therefore, any truth assignment that could satisfy formula Φ would have to assign literals x and $\neg x$ the same truth-value.
Any such truth assignment to formula Φ's *literals* would not be a valid truth assignment to Φ's *variables* (specifically to variable x).
We conclude that no valid truth assignment could satisfy Φ.

Sufficiency

Say next that $G(\Phi)$ has no strongly connected component which contains vertices arising from a variable x in both positive (x) and negated ($\neg x$) forms.

We construct a truth assignment τ to Φ's variables under which Φ evaluates to TRUE.

Assignment τ witnesses Φ's satisfiability.

We construct a satisfying truth assignment τ for Φ as follows.
1. Say that graph $G(\Phi)$ has k mutually disjoint strongly connected components.
We label these components in *topological order*, as $S_1, S_2, \ldots, S_k$, which means:

- $G(\Phi)$ contains no arc of the form $(u \to v)$ where vertex u belongs to some component S_i, and vertex v belongs to some component S_j with $j < i$.

We know that this labeling is possible because any such arc would make all vertices of S_j accessible from all vertices of S_i, and conversely.
This would mean that S_i and S_j would belong to the same strongly connected component – which would contradict the components' assumed disjointness.

2. We assign truth-values to variables of Φ by scanning the vertices/literals of $G(\Phi)$ in decreasing order of the topological indices of $G(\Phi)$'s strongly connected components.

- The *first time* that we encounter a vertex/literal ℓ, in true or negated form, we assign the truth-value to ℓ's associated variable that makes literal ℓ TRUE. This strategy also makes the clause that this instance of literal ℓ occurs in evaluate to TRUE.
- If we encounter an instance of a vertex/literal ℓ whose associated variable has already been assigned a truth-value, then we assign to this instance a truth-value that is consistent with the variable's assignment: i.e., a positive literal gets the same assignment, while a negative instance gets the negated version of the assignment.

Proceeding in this fashion, we develop a truth assignment that satisfies all of Φ's clauses.

Assume for contradiction that some clause of Φ, say $(\xi \vee \eta)$, is not satisfied under our procedure. This means, in particular, that our assignment τ assigns the truth-value FALSE to vertex/literal ξ.
But, this can happen only if t has assigned the truth-value TRUE to vertex/literal $\bar{\xi}$, within a strongly connected component of $G(\Phi)$ whose index is higher than that of the strongly connected component that ξ occurs in.
The same observation applies to vertex/literal η. But this is impossible, because within our construction of $G(\Phi)$, the clause $(\xi \vee \eta)$ in Φ would add the arc $(\bar{\xi} \rightarrow \eta)$ to graph $G(\Phi)$.
It follows that we have found a truth assignment that satisfies formula Φ, as was claimed.