

Masterarbeit

Alexander Lauer

August 23, 2022

Abstract

Contents

| | | |
|----------|--|----------|
| 1 | Preliminaries | 3 |
| 2 | partial consistency improving | 3 |
| 2.1 | partial-conditions and -satisfiability | 3 |
| 2.2 | partial consistency improving | 4 |
| 3 | application condition | 4 |

1 Preliminaries

Definition 1.1 (subgraph). Let G_1 and G_2 be graphs. The graph G_2 is called a subgraph of G_1 if an injective morphism $f : G_2 \rightarrow G_1$ exists. We use the notation $G_2 \subseteq G_1$ if G_2 is a subgraph of G_1 and $G_2 \subset G_1$ if f is not bijective. The set of all subgraphs of G_1 is denoted by $\text{sub}(G_1)$.

Definition 1.2 (partial morphism). Let $f : G_1 \rightarrow G_2$ and $g : G_3 \rightarrow G_4$ be morphisms. The morphism g is called a partial morphism of f if $G_3 \subseteq G_1$, $G_4 \subseteq G_2$ and $f|_{G_3} = g$.

Definition 1.3 (nested graph condition). A graph condition over a graph C_0 is inductively defined as follows:

- *true* is a graph condition over every graph.
- $\exists(a : C_0 \hookrightarrow C_1, d)$ is a graph condition over C_0 if a is an injective graph morphism and d is a graph condition over C_1 .
- $\neg d$ is a graph condition over C_0 if d is a graph condition over C_0 .
- $d_1 \wedge d_2$ and $d_1 \vee d_2$ are graph conditions over C_0 if d_1 and d_2 are graph conditions over C_0 .

Conditions over the empty graph \emptyset are called constraints. Every injective morphism $p : C_0 \hookrightarrow G$ satisfies *true*. An injective morphism p satisfies $\exists(a : C_0 \hookrightarrow C_1, d)$ if there exists an injective morphism $q : C_1 \hookrightarrow G$ such that $q \circ a = p$ and q satisfies c . An injective morphism satisfies $\neg d$ if it does not satisfy d , it satisfies $d_1 \wedge d_2$ if it satisfies d_1 and d_2 and it satisfies $d_1 \vee d_2$ if it satisfies d_1 or d_2 . A graph G satisfies a constraint c , $G \models c$, if $p : \emptyset \hookrightarrow G$ satisfies c . We use the abbreviation $\forall(a : C_0 \hookrightarrow C_1, d) := \neg \exists(a : C_0 \hookrightarrow C_1, \neg d)$.

The nesting level nl of a condition is defined as $\text{nl}(\text{true}) = 0$ and $\text{nl}(\exists(a : P \rightarrow Q, d)) := \text{nl}(d) + 1$.

Definition 1.4 (alternating quantifier normal form (ANF)[1]). A graph condition c is in alternating normal form (ANF) if it is of the form

$$c = Q(a_1 : C_0 \hookrightarrow C_1, \overline{Q}(a_2 : C_1 \hookrightarrow C_2, Q(a_3 : C_2 \hookrightarrow C_3, \overline{Q}(a_4 : C_3 \hookrightarrow C_4, \dots))))$$

with $Q \in \{\exists, \forall\}$ and $\overline{Q} = \exists$ if $Q = \forall$, $\overline{Q} = \forall$ if $Q = \exists$.

2 partial consistency improving

2.1 partial-conditions and -satisfiability

Definition 2.1 (partial condition). Let c be a condition over C_0 . A partial condition of c over $C'_0 \subseteq C_0$ is defined as:

1. *true* is the partial condition of *true* for every morphism.
2. if $c = \exists(a : C_0 \hookrightarrow C_1, d)$ a partial condition of c over C'_0 is given by $\exists(a' : C'_0 \hookrightarrow C'_1, d')$ with a'_1 being a partial morphism of a , $C'_1 \subseteq C_1 \setminus a(C_0 \setminus C'_0)$ and d' is a partial condition of d over C'_1 .
3. if $c = \forall(a : C_0 \hookrightarrow C_1, d)$ a partial condition of c over C'_0 is given by $\forall(a' : C'_0 \hookrightarrow C'_1, d')$ with a'_1 being a partial morphism of a , $C'_1 = C_1 \setminus a(C_0 \setminus C'_0)$ and d' is a partial condition of d over C'_1 .
4. if $c = d_1 \wedge d_2$ or $c = d_1 \vee d_2$ the partial condition of c is given by $d'_1 \wedge d'_2$ and $d'_1 \vee d'_2$, respectively, with d'_1 and d'_2 being partial conditions of d_1 and d_2 over C'_0 .
5. if $c = \neg d$ the partial condition of c is given by $\neg d'$ with d' being a partial condition of d over C'_0 .

We use the notation $c' \leq c$ if c' is partial condition of c over $C'_0 \subseteq C_0$ and $c' < c$ if $C'_0 \subset C_0$.

Definition 2.2 (partial satisfiability). Let a condition c over C_0 and a graph G be given. A morphism $p_0 : C_0 \rightarrow G$ partial satisfies c , $p_0 \models_p c$ if a partial morphism $p'_0 : C'_0 \rightarrow G$ of p_0 satisfies a partial condition of c over C'_0 .

Note, that $p_0 \models c$ implies $p_0 \models_p c$

2.2 partial consistency improving

Definition 2.3 (improving atomic transformation). Given a graph G and a constraint c . An atomic transformation is a graph transformation that inserts or deletes one single element. A set of atomic transformations is called improving set if the graph derived by application of all transformations in A satisfies c . A is called a minimal improving set if no improving set $B \subset A$ exists. Let I be the set of all minimal improving sets for G regarding c . The total number of improving atomic transformations for G regarding c is given by

$$\text{iat}(G, c) := \left| \bigcup_{A \in I} A \right|$$

Definition 2.4 (partial consistency improving transformation). Given a constraint c and a rule r , a transformation $G \xrightarrow{r, m} H$ is partial consistency improving with respect to c if $\text{iat}(H, c) < \text{iat}(G, c)$.

Construction 2.5. Let a graph G and a constraint c be given.

3 application condition

Construction 3.1. Given a rule $r = L \xleftarrow{l} K \xrightarrow{r} R$ and a constraint c in ANF with $\text{nl}(c) = 1$. Let G and H be graphs. We denote the set of all overlaps of G and H as $\text{ol}(G, H)$. We construct the following application conditions:

1. if $c = \forall(a_0 : C_0 \rightarrow C_1, \text{false})$:

$$\bigvee_{Q \in \text{ol}(L, C_1)} \left(\exists(a : L \hookrightarrow Q, \text{true}) \wedge \text{Left}(\neg \exists(b : R \rightarrow Q, \text{true})) \right)$$

2. if $c = \exists(a_0 : C_0 \rightarrow C_1, \text{true})$:

$$\bigvee_{Q \in \text{ol}(L, C_1)} \bigvee_{Q' \in \text{sub}(Q)} \left(\exists(a : L \rightarrow Q', \text{true}) \wedge \left(\bigvee_{\substack{\bar{Q} \in \text{sub}(Q) \\ \bar{Q} \subset Q'}} \text{Left}(\exists(a : R \rightarrow \bar{Q}, \text{true})) \right) \right)$$

References

- [1] C. Sandmann and A. Habel. [Rule-based graph repair](#). *arXiv preprint arXiv:1912.09610*, 2019.