Verification of Graph Transformation Systems with Whole Neighbourhood Operations

Giorgio Delzanno\textsuperscript{1}    \textbf{Jan Stückrath}\textsuperscript{2}

\textsuperscript{1}Università di Genova, Italy
\textsuperscript{2}Universität Duisburg-Essen, Germany
Motivation

**Overall Goal:**
Provide an high-level automatic decision procedure for the correctness of systems, e.g. protocols.
Motivation

**Overall Goal:**
Provide an high-level automatic decision procedure for the correctness of systems, e.g. protocols.

In [CONCUR 2014] we provided a general framework for the verification of graph transformation systems:

- Model states and behaviour of the system by graphs and graph transformation rules.
- Specify sets of erroneous states.
- Automatically check if an erroneous state is reachable.

⇝ Undecidable in general.
Motivation

**Overall Goal:**
Provide an high-level automatic decision procedure for the correctness of systems, e.g. protocols.

In [CONCUR 2014] we provided a general framework for the verification of graph transformation systems:

- Model states and behaviour of the system by graphs and graph transformation rules.
- Specify sets of erroneous states.
- Automatically check if an erroneous state is reachable.

Undecidable in general.

This talk is about extending the framework with an alternative rewriting formalism (accepted at RP 2014).
Hypergraphs

Definition (Hypergraph)

A $\Lambda$-hypergraph consists of a set of nodes $V$, a set of edges $E$, a connection function $c : E \rightarrow V^*$ and a labelling function $l : E \rightarrow \Lambda$. The length of the node sequence is determined by the label.


**Example - Dining Philosophers**

**Dining Philosophers**, similar to [Namjoshi, Tefler]:
- A philosopher can be hungry ($H$), eating ($E$) or thinking ($T$).
- Network structure is arbitrary.
- Forks can be free ($F$) or owned ($OF$).
- To eat a philosopher has to own all forks "within reach".
- Can a situation be reached, where two philosophers sharing a fork eat at the same time?

![Graph Diagram]
Example - Dining Philosophers
Example - Dining Philosophers
Example - Dining Philosophers

![Diagram of the Dining Philosophers problem with graph transformation analysis.]
Example - Dining Philosophers
Example - Dining Philosophers

\[ H \rightarrow_{OF} E \rightarrow_{OF} H \]

\[ H \rightarrow_{OF} F \rightarrow_{OF} H \]

\[ H \rightarrow_{OF} F \rightarrow_{OF} H \]

\[ H \rightarrow_{OF} E \rightarrow_{OF} H \]
Single Pushout Approach

As rewriting formalism we use SPO with partial morphisms \( r : L \rightarrow R \) as rules and total, injective matches.

Graph transformation rule

Transition step
Universal Quantification

Definition (Universally quantified rules)

A universally quantified rule is a pair \((r, U)\), where \(r : L \rightarrow R\) is a partial morphism and \(U\) is a finite set of universal quantifications.

A universal quantification is a pair \((p_u, q_u) \in U\) where:

- \(p_u : L \rightarrow L_u\) is a total injective morphism
- \(q_u : L_u \rightarrow R_u\) is a partial morphism which is injective on \(p_u(L)\)
Instantiation - Example

\[ R_u \xrightarrow{q_u} L_u \xleftarrow{p_u} \]

\[ \xrightarrow{r} \]

\[ F \quad OF \quad E \quad T \]
Instantiation - Example

Graph Transformation Analysis Implementation and Conclusion

Overview

\[ R_u \]

\[ q_u \]

\[ L_u \]

\[ p_u \]

\[ G \]

\[ H \]

\[ OF \]

\[ F \]

\[ E \]

\[ L \]

\[ E \]

\[ R \]

\[ T \]

\[ m \]

\[ r \]
Instantiation - Example

\[ R_u \quad q_u \quad L_u \quad p_u \quad G \quad H \quad F \quad E \quad OF \quad T \quad r \quad m \quad \bar{m} \quad \eta \]
Instantiation - Example

1. $E \xrightarrow{r_0} T$
2. $OF \xrightarrow{r_1} F$
3. $OF \xrightarrow{r_2} F$
Instantiation - Example

\[
L \xrightarrow{id_L} \xrightarrow{r} R
\]

\[
L_u \xrightarrow{p_u} R_u \xrightarrow{q_u}
\]

\[
\begin{align*}
L & \quad E \\
& \quad OF \\
R & \quad T \\
& \quad F
\end{align*}
\]
Instantiation - Example
Instantiation - Example

\[ L \rightarrow id_L \rightarrow L \rightarrow r \rightarrow R \]

\[ L_u \rightarrow id'_L \rightarrow \bar{L}_u \rightarrow \eta \rightarrow \bar{R}_u \]

\[ \begin{align*}
L & \quad E \\
R & \quad T \\
\bar{L}_u & \quad E \\
\bar{R}_u & \quad T \\
\end{align*} \]

\[ \begin{align*}
L & \quad E \\
R & \quad T \\
\bar{L}_u & \quad E \\
\bar{R}_u & \quad T \\
\end{align*} \]
Definition (Rule application)

Let $\rho$ be a universally quantified rule. We say that $\rho$ is applicable to a graph $G$, if there is an instantiation $\eta$ and a match $m$, such that the neighbourhood of every quantified node is matched.
How to analyse such systems, while retaining an infinite state space?

We use the theory of well-structured transition systems [CONCUR 2014], for which coverability is decidable [Finkel, Schnoebelen], [Abdulla et al.].
Coverability Problem

Let $\mathcal{T} = (S, \Rightarrow)$ be a transition system, $s_0$ the initial state and $\leq$ an order on states.

Coverability Problem

Given $s_f$, is there a state $s'$ such that $s_0 \Rightarrow^* s'$ and $s_f \leq s'$?

$s_f$ is coverable
Subgraphs for Coverability

**Definition (Subgraph)**

A graph $G$ is a subgraph of a graph $G'$, if $G$ can be obtained from $G'$ by a sequence of node deletions and edge deletions.
Subgraphs for Coverability

Definition (Subgraph)

A graph $G$ is a subgraph of a graph $G'$, if $G$ can be obtained from $G'$ by a sequence of node deletions and edge deletions.

Initial Problem:
Is one of the following graphs coverable?
Condition 1: Well-quasi-order

Definition (Well-quasi-order)

A reflexive, transitive relation $\leq$ is a well-quasi-order if:

- In every infinite sequence $x_0, x_1, x_2, x_3, \ldots$ there exist indices $i < j$ such that $x_i \leq x_j$. 
Definition (Well-quasi-order)

A reflexive, transitive relation $\leq$ is a well-quasi-order if:

- In every infinite sequence $x_0, x_1, x_2, x_3, \ldots$ there exist indices $i < j$ such that $x_i \leq x_j$.

The subgraph ordering is

- not a well-quasi-order on all graphs.

- a well-quasi-order on the set of graphs $\mathcal{G}_k$ with bounded longest (undirected) paths [Ding][Meyer].
Condition 2: Monotonicity

Definition (Monotonicity)

A graph transformation system is monotone if:
whenever $G_1 \subseteq H_1$ and $G_1 \Rightarrow G_2$, there exists a $H_2$ such that $H_1 \Rightarrow^* H_2$ and $H_2 \subseteq G_2$. 

$$
\begin{align*}
H_1 & \Rightarrow^* H_2 \\
G_1 & \Rightarrow G_2
\end{align*}
$$
Monotonicity and Subgraphs

Every GTS without negative application conditions is naturally a $Q$-restricted WSTS.
Monotonicity and Subgraphs

Every GTS without negative application conditions is naturally a $Q$-restricted WSTS.
Monotonicity and Subgraphs

Every GTS without negative application conditions is naturally a $Q$-restricted WSTS.

However, universally quantified rules may violate the compatibility condition.

leads to over-approximation.
Solving Coverability by a Backward Search

We use the well-known backward search for WSTS:

- It computes the set of minimal representatives $\mathcal{W}$ of all graphs that can cover an error.
- If there is no $G' \in \mathcal{W}$ with $G' \subseteq G$, then no error graph is coverable (within $\Rightarrow_{G_k}$) from $G$.
- We can use $\Rightarrow$ instead of $\Rightarrow_{G_k}$, but lose the guarantee of termination.
In [CONCUR 2014] we showed how a backward step can be computed.
However, a rule can have infinitely many instantiations!

**Proposition**

In each backward step, only finitely many instantiations have to be applied backwards.

~~> For each rule and graph there is a constant, such that results for instantiations of at least that size are subsumed by results for smaller instantiations.
We integrated the presented formalism in the Tool UNCOVER. It takes \( \sim 1 \) second to compute 12 minimal errors for the Dining Philosophers example.
Ongoing and Future Work

Related work:

- Adaptive star grammars. [Drewes at al.]
  ~⇒ Our formalisms can be seen as an extension.
- Technical similarities with amalgamation.
  [Boehm, Fonio, Habel]

Future work:

- Is our over-approximation precise enough? How can we increase precision?
- Can this approach be transferred to other orders, e.g. minors?
- With which order are universally quantified rules monotone?
- Use graph patterns to represent set of graphs?
  cf. [Saksena, Wibling, Jonsson]
Thank you for your attention!

Any Questions?