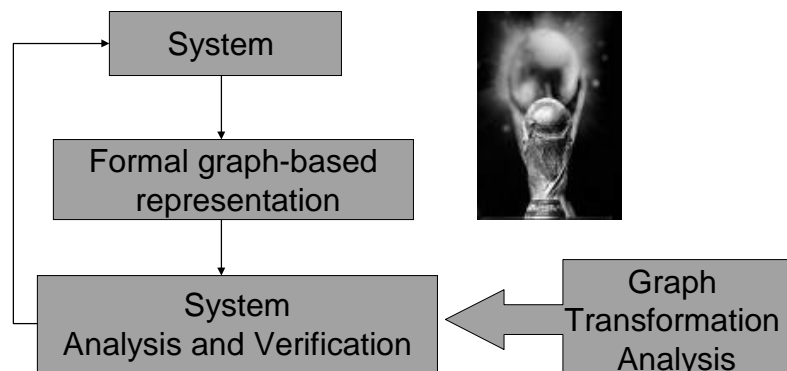


Conflict Detection for Graph Transformation with Negative Application Conditions

Leen Lambers, Hartmut Ehrig, Fernando Orejas



Analysis of Graph Transformation



Graph Transformation and its
Analysis Techniques

2

Analysis of Graph Transformation in Software Engineering

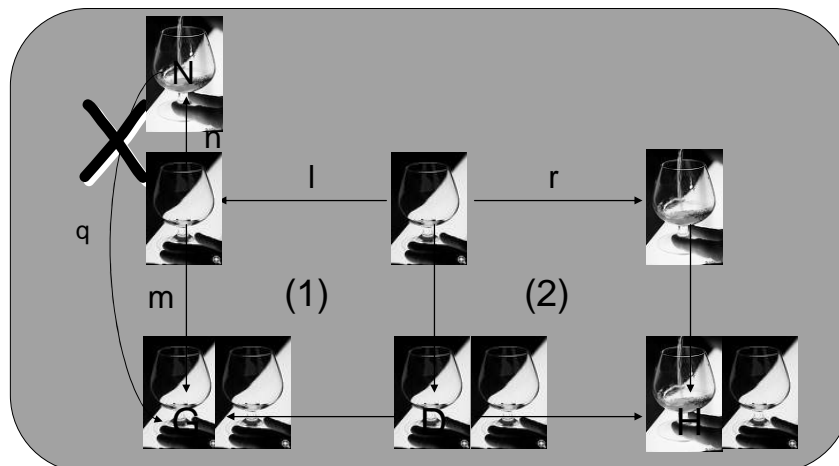
- Visual Languages:
 - *Efficient Parsing of Visual Languages based on Critical Pair Analysis and Contextual Layered Graph Transformation*, Bottoni, P., A. Schürr and G. Taentzer, 2000
- Model transformation:
 - *Automated Model Transformation and its Validation using AToM 3 and AGG*, de Lara, J. and G. Taentzer, 2004
- Security:
 - *Graph-based Specification of Access Control Policies* Koch, M., Mancini, L., Parisi-Presicce, F., 2005
- Refactoring:
 - *Detecting Structural Refactoring Conflicts using Critical Pair Analysis*, Mens, T., G. Taentzer and O. Runge, 2004
- Testing:
 - *Automatic Conformance Testing of Web Services*, R. Heckel and L. Mariani, 2005
- Aspect-Oriented Programming
 - *Interaction Analysis in Aspect-Oriented Models*, K. Mehner, M. Monga and G. Taentzer



Graph Transformation and its
Analysis Techniques

3

Graph Transformation with NACs



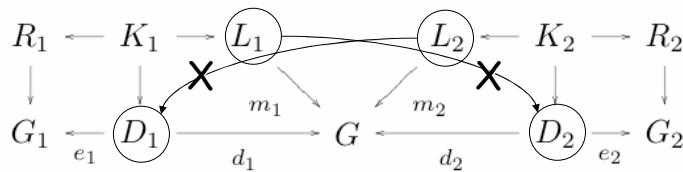
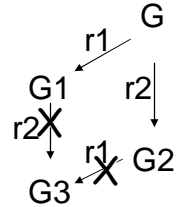
Graph Transformation and its
Analysis Techniques

4

Analysis of Graph Transformation Systems

without NACs

- parallel independence
- conflict



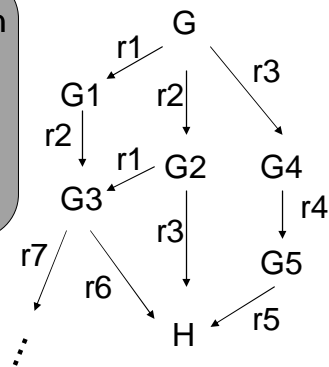
Graph Transformation and its Analysis Techniques

5

Analysis of Graph Transformation Systems

without NACs

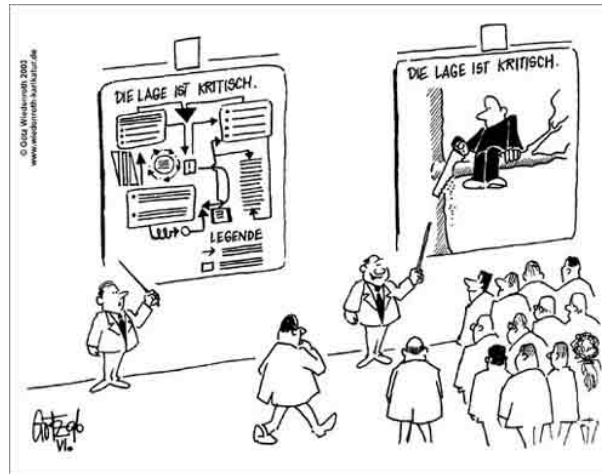
- Conflict and Dependency Detection
- Confluence Analysis
- Termination Analysis



Graph Transformation and its Analysis Techniques

6

Conflict Detection



Graph Transformation and its Analysis Techniques

7

without NACs

Conflict in Minimal Context

A critical pair is a pair of direct transformations $P_1 \leftarrow K \Rightarrow P_2$ in conflict, s.t. m_1 and m_2 are jointly surjective morphisms.

$$\begin{array}{ccccccc}
 R_1 & \leftarrow & K_1 & \rightarrow & L_1 & & L_2 & \leftarrow & K_2 & \rightarrow & R_2 \\
 \downarrow & & \downarrow & & \searrow & & \swarrow & & \downarrow & & \downarrow \\
 P_1 & \xleftarrow{e_1} & D_1 & \xrightarrow{d_1} & K & \xleftarrow{d_2} & D_2 & \xrightarrow{e_2} & P_2 \\
 & & & \nearrow m_1 & & \nwarrow m_2 & & & & &
 \end{array}$$



Graph Transformation and its Analysis Techniques

8

without NACs

Conflict Detection

conflict
minimal context

=


critical pair

completeness theorem

:

conflict

bigger context



Graph Transformation and its Analysis Techniques

9

without NACs

Conflict Detection

conflict
minimal context

=


critical pair

completeness theorem

:

critical pair

bigger context



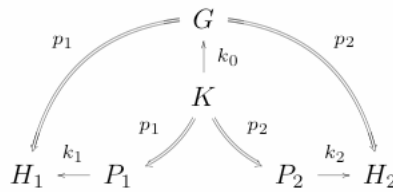
Graph Transformation and its Analysis Techniques

10

without NACs

Conflict Detection

Completeness of critical pairs: For each pair of direct transformations $H_1 \leftarrow G \Rightarrow H_2$ in conflict, there exists a critical pair $P_1 \leftarrow K \Rightarrow P_2$ expressing the same conflict in a minimal context.



without NACs

Conflict Detection



=



completeness theorem

:

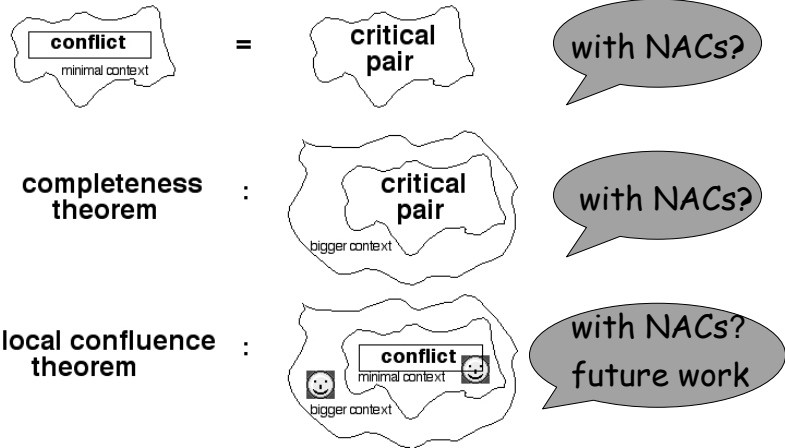


local confluence theorem

:



Conflict Detection

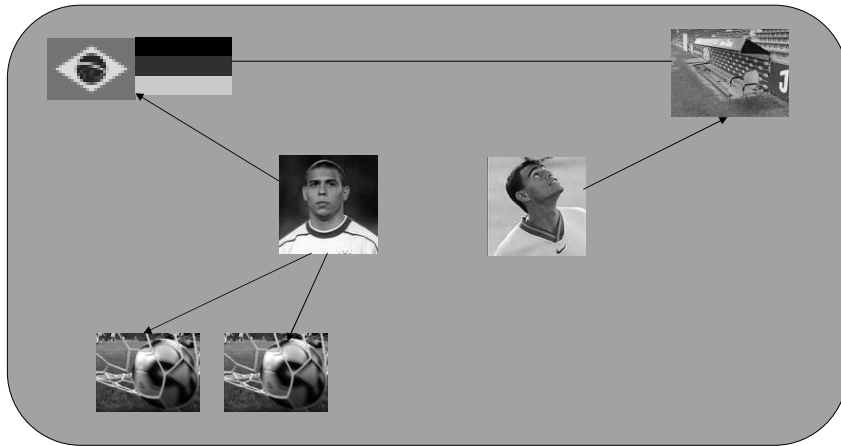


World Cup and Graph Transformation



without NACs

Delete / Use Conflict

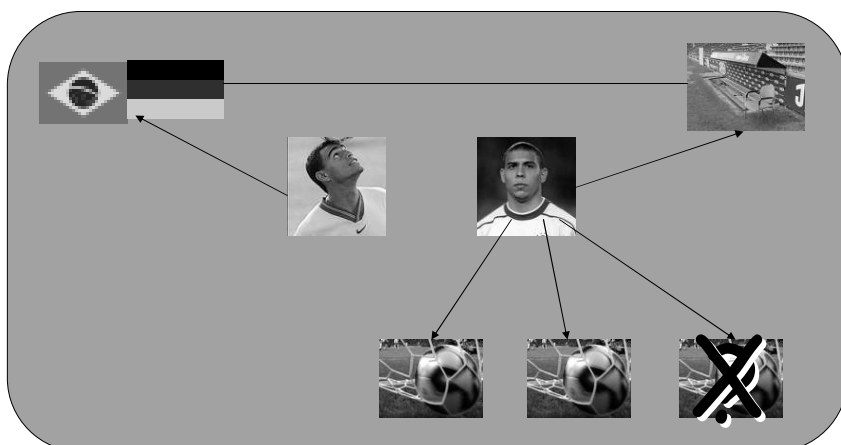


Graph Transformation and its
Analysis Techniques

15

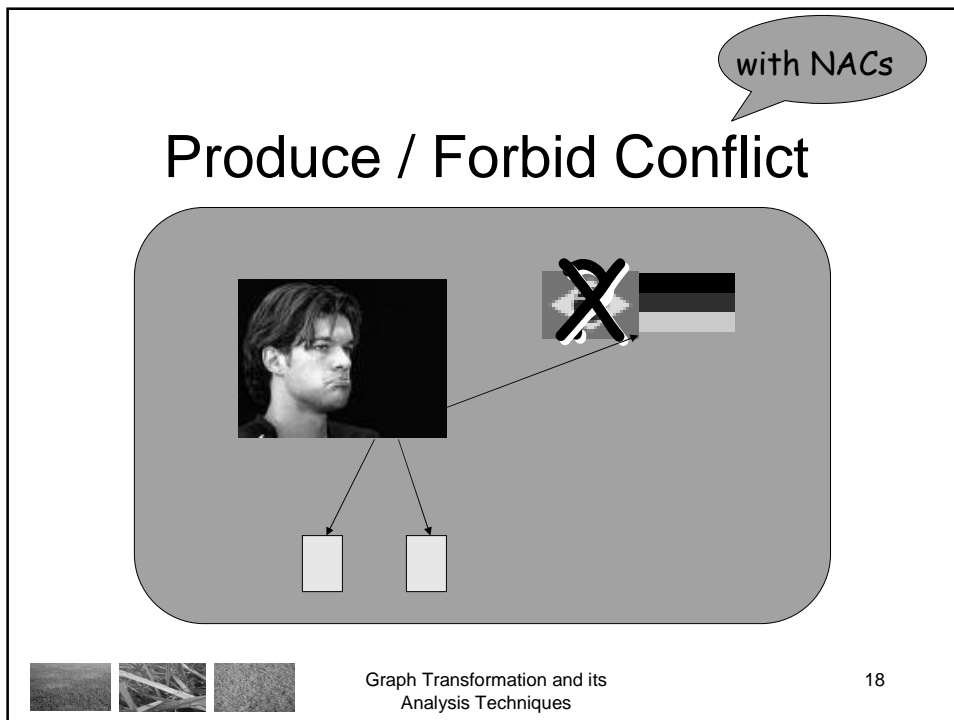
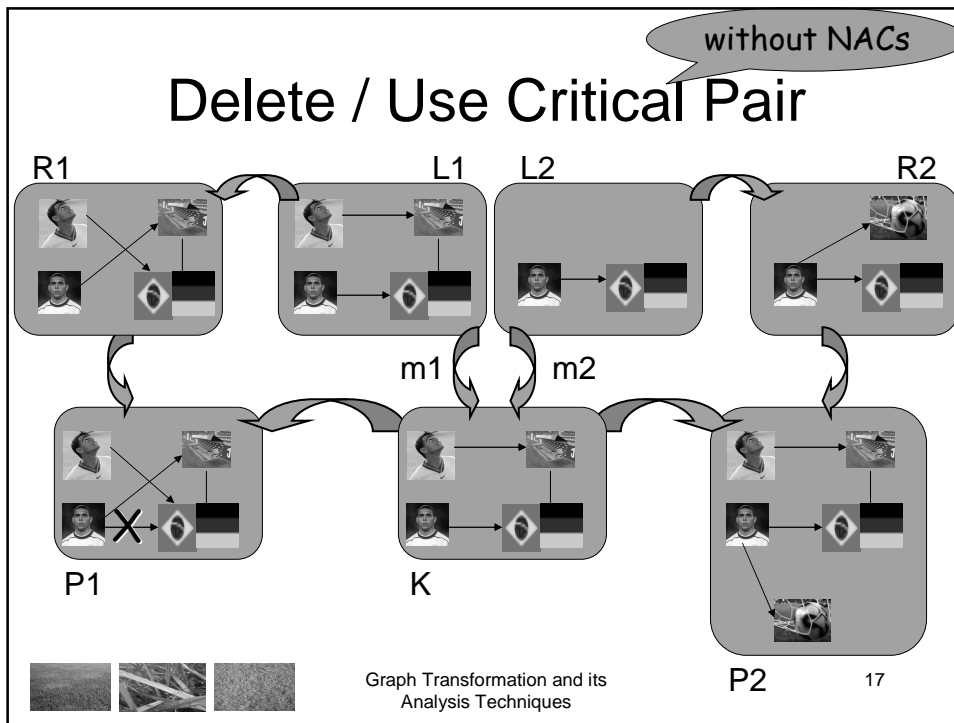
without NACs

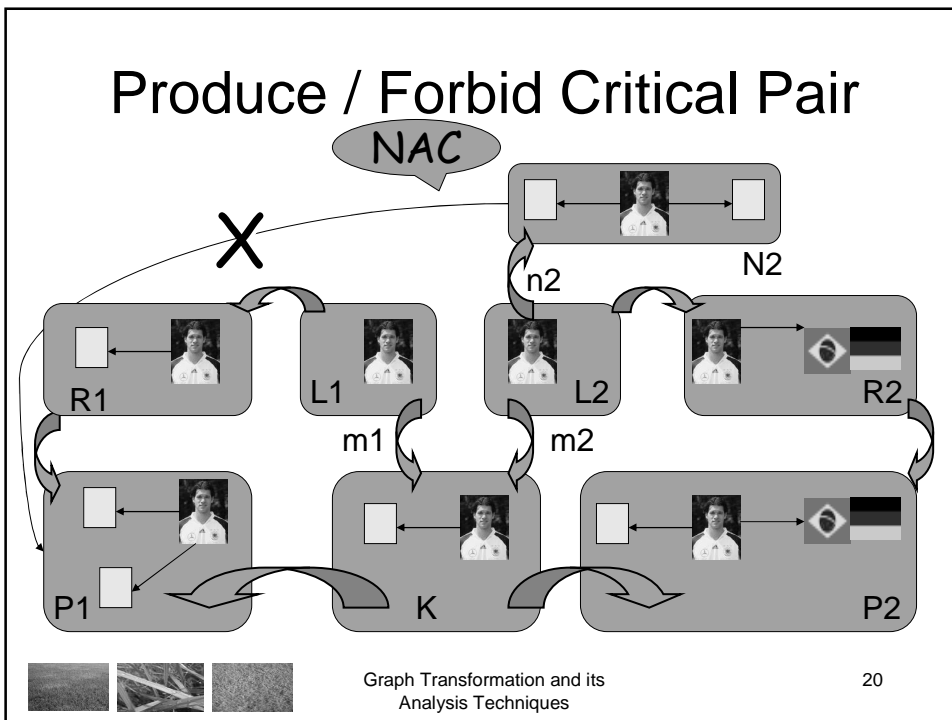
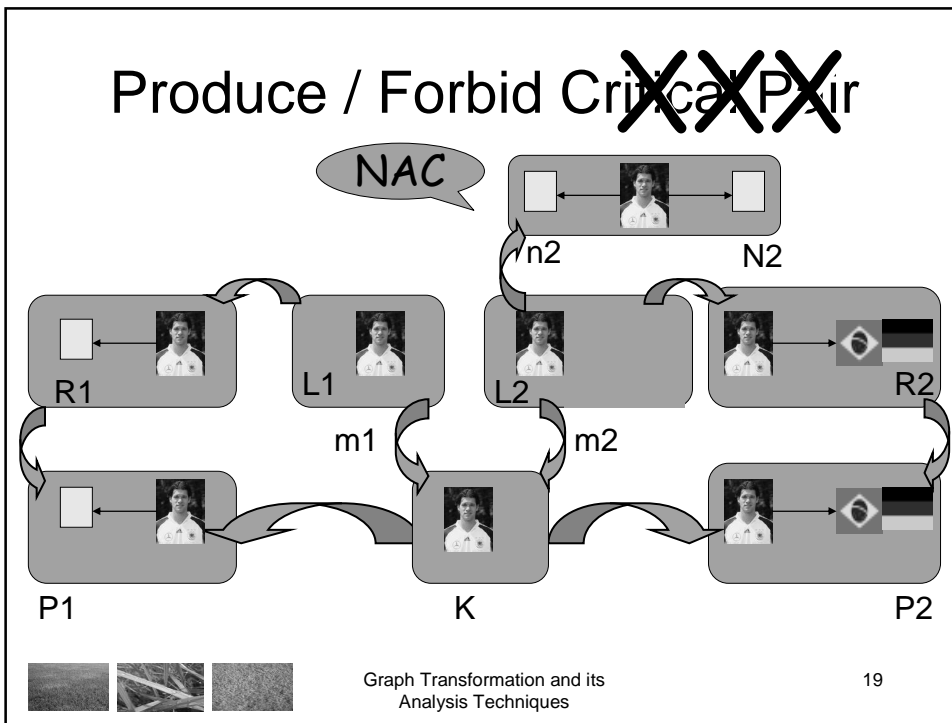
Delete / Use Conflict



Graph Transformation and its
Analysis Techniques

16

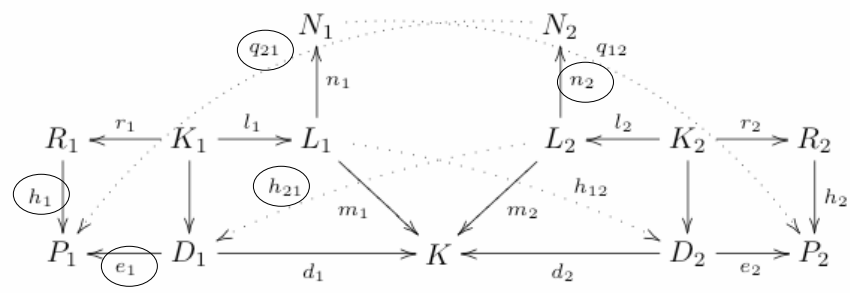




with NACs

Produce/Forbid Critical Pair

(b) there exists a $h_{21} : L_2 \rightarrow D_1$ s.t. $d_1 \circ h_{21} = m_2$, but for one of the NACs $n_2 : L_2 \rightarrow N_2$ of p_2 there exists a morphism $q_{21} : N_2 \rightarrow P_1 \in M$ s.t. $q_{21} \circ n_2 = e_1 \circ h_{21}$ and (q_{21}, h_1) jointly surjective (produce-forbid-conflict)

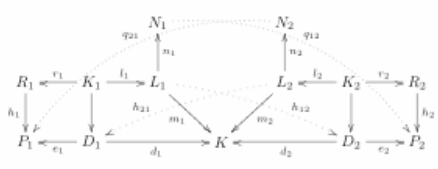


Conflict in Minimal Context

Definition 8 (critical pair). A critical pair is a pair of direct transformations $K \xrightarrow{(p_1, m_1)} P_1$ with NAC_{p_1} and $K \xrightarrow{(p_2, m_2)} P_2$ with NAC_{p_2} such that:

1. (a) $\exists h_{12} : L_1 \rightarrow D_2 : d_2 \circ h_{12} = m_1$ and (n_1, m_2) jointly surjective (use-delete-conflict)
or
(b) there exists a $h_{12} : L_1 \rightarrow D_2$ s.t. $d_2 \circ h_{12} = m_1$, but for one of the NACs $n_1 : L_1 \rightarrow N_1$ of p_1 there exists a morphism $q_{12} : N_1 \rightarrow P_2 \in M$ s.t. $q_{12} \circ n_1 = e_2 \circ h_{12}$ and (q_{12}, h_2) jointly surjective (forbid-produce-conflict)
- or
2. (a) $\exists h_{21} : L_2 \rightarrow D_1 : d_1 \circ h_{21} = m_2$ and (n_1, m_2) jointly surjective (delete-use-conflict)
or
(b) there exists a $h_{21} : L_2 \rightarrow D_1$ s.t. $d_1 \circ h_{21} = m_2$, but for one of the NACs $n_2 : L_2 \rightarrow N_2$ of p_2 there exists a morphism $q_{21} : N_2 \rightarrow P_1 \in M$ s.t. $q_{21} \circ n_2 = e_1 \circ h_{21}$ and (q_{21}, h_1) jointly surjective (produce-forbid-conflict)

with NACs



Summary

Graph Transformation with NACs

- Conflict Characterization
- Critical Pair Definition
- Completeness Theorem
- Critical Pair Construction
- Future Work

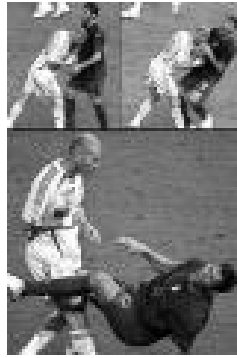


Tool Support

- The Attributed Graph Grammar System AGG:
A Development Environment for Attributed Graph Transformation Systems
- Conflict Detection with AGG
- <http://tfs.cs.tu-berlin.de/agg/>



Questions to Conflicts ?



Graph Transformation and its
Analysis Techniques

25