Algebraic Graph Transformation: NACs and Graph Constraints

Fernando Orejas
Royal Holloway University of London
on leave from Universitat Politècnica de Catalunya, Barcelona
1. Introduction
2. NACs
3. Parallel Independence and Critical Pairs for Rules with NACs
4. Graph Constraints
5. Modelling and Reasoning with Constraints
Introduction
NACs and Constraints
Negative Application Conditions
(Negative) Application Conditions (NACs)

Given a rule

\[ p = L \xleftrightarrow{K} R \]

a negative application condition is an inclusion

\[ L \rightarrow C \quad \text{or} \quad L \rightarrow X \rightarrow C \]

or

\[ R \rightarrow C \quad \text{or} \quad R \rightarrow X \rightarrow C \]
NAC satisfaction:

\[(h: L \to G) \models L \to C\]

there is no \(f: C \to G\) such that the diagram commutes

\[
\begin{array}{ccc}
C & \xleftarrow{n} & L \\
\downarrow{f} & & \downarrow{h} \\
C & & G
\end{array}
\]
NAC satisfaction:

\[(h: L \rightarrow G) \models L \rightarrow X \rightarrow C\]

if for every \(f: X \rightarrow G\) such that the left triangle commutes there is a \(g: C \rightarrow G\) such that the right triangle commutes
Graph Transformation with NACs

Given the rule

\[ p = L \xleftarrow{\text{l}} K \xrightarrow{\text{r}} R \]

together with NACs (\(N_L, N_R\)), we can apply \(p\) to \(G\) via \(m\): if \(m\) satisfies all the NACs in \(N_L\) and \(m'\) satisfies all the NACs in \(N_R\).
Right NACs elimination

If $R \rightarrow C$ is a right NAC, there is a left NAC $L \rightarrow C'$ such that:

$m$ satisfies $L \rightarrow C'$ if and only if $m'$ satisfies $R \rightarrow C$.

\[
\begin{array}{ccccccc}
    & L & \leftarrow & K & \rightarrow & R \\
    m & \downarrow & & \downarrow & & \downarrow & m' \\
    G & \leftarrow & D & \rightarrow & H
\end{array}
\]
Right NACs elimination (Proof)

Construction:

If there is a pushout complement $D'$
Right NACs elimination
Right NACs elimination
Parallel Independence and Critical Pairs for Rules with NACs
Parallel Independence
Parallel Independence with NACs

Two transformations with NACs on the same graph $G$ are parallel independent:

If there exist morphisms $h_1$ and $h_2$ such that $d_2 \cdot h_1 = m_1$, $d_1 \cdot h_2 = m_2$, $e_1 \cdot h_2 \vdash N_2$, and $e_2 \cdot h_1 \vdash N_1$
A critical pair consists of two transformations with NACs on the same graph $K$ such that $m_1$ and $m_2$ are jointly surjective and one of the following conditions hold:

1. There is no morphism $h_1$ such that $d_2 \cdot h_1 = m_1$
There is no morphism $h_2$ such that $d_1 \cdot h_2 = m_2$

There is a morphism $h_1$ but $e_2 \cdot h_1 \models N_1$

There is a morphism $h_2$ but $e_1 \cdot h_2 \models N_2$
Completeness Theorem

For any two parallel dependent direct transformations with NACs:

\[ G_1 \xleftarrow{p_1,m_1} G' \xrightarrow{p_2,m_2'} G_2 \]

there is a critical pair \( P_1 \xleftarrow{p_1,m_1} G \xrightarrow{p_2,m_2} P_2 \) such that:

\[ P_1 \xleftarrow{p_1,m_1'} G' \xrightarrow{p_2,m_2'} G_2 \]
Local Confluence Theorem

A graph transformation system with NACs is locally confluent if all its critical pairs are strictly NAC-confluent.
Elimination of conflicts

Given the critical pair $P_1 \overset{\text{p}_1,\text{m}_1}{\Leftarrow} K \Rightarrow_{\text{p}_2,\text{m}_2} P_2$, if we know that, in the situation where the conflict happens, the right rule to be applied is $p_1$, then the conflict can be eliminated by adding to $p_2$ the filter $\text{NAC} \ L_2 \rightarrow K$. 
Generalization

The previous results have been generalized to arbitrary M-adhesive categories where, in addition, NACs are arbitrary conditions written in the Logic of Nested Conditions (A. Habel and K.H Penneman (2009)), as expressive as First-Order Logic.
Graph Constraints
Two kinds of constraints:

**Basic Constraints**
- \( C \)

**Conditional Constraints**
- \((c: X \rightarrow C)\)
Two kinds of constraints:

**Basic Constraints**
- C

**Conditional Constraints**
- \((c: X \rightarrow C)\)

Basic constraints are a special case of conditional constraints (when \(X = \emptyset\))
Two kinds of constraints:

Basic Constraints
- C

Conditional Constraints
- (c: X → C)

plus ¬, ∧, ∨

Basic constraints are a special case of conditional constraints (when X = ∅)
Satisfaction of basic constraints:

\[ G \models C \quad \text{if} \quad h: C \rightarrow G \]

Satisfaction of conditional constraints:

\[ G \models (g:X \rightarrow C) \quad \text{if for every} \quad h: C \rightarrow G, \quad \text{there is an} \quad f: C \rightarrow G \]

\text{such that:}

\[
\begin{array}{c}
G \\
h \\
X \\
g \\
f \\
C
\end{array}
\]
Examples

If  ¬  then

¬
Graph Transformation with constraints

Given a set of constraints $C$, and given

$$p = L \leftarrow K \xrightarrow{r} R$$

we can apply $p$ to $G$ if

$$H \models C.$$
Transformation of constraints into NACs

Given a constraint $c$, there is a set of right NACs $\Phi$ such that
given:

$m'$ satisfies all the NACs in $\Phi$ if and only if $H$ satisfies $c$. 
If $c = \neg C$, then $\Phi$ is the set of right NACs $f: R \to H$:

\[
\begin{array}{c}
\text{Construction} \\
\text{If } c = \neg C, \text{ then } \Phi \text{ is the set of right NACs } f: R \to H:\n\end{array}
\]

such that $f$ and $g$ are jointly surjective.
Proof

Let us assume that

\[ \text{Diagram:} \]

- **L** \( \overset{m}{\longrightarrow} \) **G** 
- **K** \( \overset{r}{\longrightarrow} \) **R** 
- **D** \( \overset{m'}{\longrightarrow} \) **H** 
- **L** \( \overset{l}{\longleftarrow} \) **K** 
- **G** \( \overset{m}{\longleftarrow} \) **D** 
- **H** \( \overset{m'}{\longleftarrow} \) **C**
Proof

By pair factorization there are $C'$, $f$, and $g$ such that $f$ and $g$ are jointly surjective
Example
Modelling and Reasoning with Constraints
Modelling with Constraints

Example

(1) \( \exists \) subject
\[ \text{name} = \text{CS1} \]

(2) \( \exists \) subject
\[ \text{name} = \text{CS2} \]

(3) If
\[ \text{name} = N \]

then
\[ \text{name} = N \]

\( \text{lecturer} \)
\( \text{room} \)
(6) \[ \exists \text{ lecturer} \rightarrow \begin{cases} \text{ subject } \\ \text{ room} \end{cases} \]

(7) \[ \neg \text{ If } \begin{cases} \text{ lecturer} \\ \text{ name } = N \end{cases} \text{ then } \begin{cases} \text{ lecturer} \\ \text{ name } = N \end{cases} \rightarrow \text{ subject} \]
Reasoning with constraints

1)

\[ \exists C_1 \lor \Gamma_1 \quad \neg \exists C_2 \]

\[ \therefore \Gamma_1 \quad \text{if there is a morphism } C_2 \to C_1 \]

2)

\[ \exists C_1 \lor \Gamma_1 \quad \exists C_2 \]

\[ \therefore \exists G_1 \lor \ldots \lor \exists G_k \lor \Gamma_1 \]

\[ \text{j.surj.} \]

C1 \quad \rightarrow \quad C2

Gi
3)

\[ \exists C_1 \lor \Gamma \quad \forall (g':X \rightarrow C_2) \]
\[ \exists G_1 \lor \ldots \lor \exists G_k \lor \Gamma \]

If \( X \rightarrow C_1 \) and:

- \( X \rightarrow C_2 \)
- \( X \rightarrow C_1 \)
- \( X \rightarrow C_1 \rightarrow C_2 \)
- \( X \rightarrow C_1 \rightarrow G_i \)
Example

\[
\begin{align*}
(1) & \exists \ & \text{subject} \\
& \quad name = \text{CSI} \\
(2) & \exists \ & \text{subject} \\
& \quad name = \text{CSI2}
\end{align*}
\]

\[
\begin{align*}
\exists C_1 \lor \Gamma_1 \quad \exists C_2 \\
\exists G_1 \lor \cdots \lor \exists G_k \lor \Gamma_1
\end{align*}
\]
(3) \[ \text{If } \quad \begin{array}{c} \text{subject} \\ name = N \end{array} \quad \text{then} \quad \begin{array}{c} \text{subject} \\ name = N \end{array} \]

(8) \[ \exists \quad \begin{array}{c} \text{subject} \\ name = CS1 \end{array} \quad \exists \quad \begin{array}{c} \text{subject} \\ name = CS2 \end{array} \]

\[ \exists C_1 \lor \Gamma \quad \forall (g:X \rightarrow C_2) \]

\[ \exists G_1 \lor \ldots \lor \exists G_k \lor \Gamma \]

If \( X \rightarrow C_1 \) and:

\[ X \quad \rightarrow \quad C_2 \]

\[ C_1 \quad \rightarrow \quad Gi \quad \text{j.surj.} \]
\[ \exists C_1 \lor \Gamma_1 \not\exists C_2 \]

if there is a morphism $C_2 \rightarrow C_1$
∃V
¡Thank You!