Checking Graph-Transformation Systems for Confluence

Detlef Plump
Hypergraphs
Hypergraphs

Signature $\Sigma = \langle \Sigma_V, \Sigma_E, \text{Type: } \Sigma_E \rightarrow \mathcal{P}(\Sigma^*_V) \rangle$ where $\Sigma_V$ and $\Sigma_E$ are sets of vertex labels and hyperedge labels
Hypergraphs

- **Signature** $\Sigma = \langle \Sigma_V, \Sigma_E, \text{Type}: \Sigma_E \rightarrow \mathcal{P}(\Sigma_V^*) \rangle$ where $\Sigma_V$ and $\Sigma_E$ are sets of vertex labels and hyperedge labels.

- **Hypergraph**
  $G = \langle V, E, l_V: V \rightarrow \Sigma_V, l_E: E \rightarrow \Sigma_E, \text{att}: E \rightarrow V^* \rangle$ where $V, E$ are finite sets of vertices and hyperedges, and $l_V^*(\text{att}(e)) \in \text{Type}(l_E(e))$ for each hyperedge $e$. 

---

[Diagram showing data and stack with push and pop operations, and connections to boolean values indicating top, empty, true, and false.]
Hypergraphs

- **Signature** \( \Sigma = \langle \Sigma_V, \Sigma_E, \text{Type}: \Sigma_E \to \mathcal{P}(\Sigma_V^*) \rangle \) where \( \Sigma_V \) and \( \Sigma_E \) are sets of vertex labels and hyperedge labels.

- **Hypergraph**
  \( G = \langle V, E, l_V: V \to \Sigma_V, l_E: E \to \Sigma_E, \text{att}: E \to V^* \rangle \) where \( V, E \) are finite sets of vertices and hyperedges, and \( l_V^*(\text{att}(e)) \in \text{Type}(l_E(e)) \) for each hyperedge \( e \).

- **Graph** \( G \): all hyperedges \( e \) satisfy \( |\text{att}(e)| = 2 \).
Rules (injective case)

**Rule:** \( \langle L \supseteq K \subseteq R \rangle \)
Rules (injective case)

- **Rule:**  $\langle L \supseteq K \subseteq R \rangle$

  ![Diagram]

  - **Intuition:** remove $L - K$ and add $R - K$
Rules (injective case)

- **Rule:** \(\langle L \supseteq K \subseteq R \rangle\)

- **Intuition:** remove \(L - K\) and add \(R - K\)

- **Short notation:** \(L \Rightarrow R\)
Direct derivation $G \Rightarrow_{r,g} H$

Given $r: \langle L \supseteq K \subseteq R \rangle$ and injective morphism $g: L \rightarrow G$
Direct derivation $G \Rightarrow_{r,g} H$

Given $r$: $\langle L \supseteq K \subseteq R \rangle$ and injective morphism $g$: $L \rightarrow G$

1. Check **dangling condition**: nodes in $g(L - K)$ are not incident to edges outside $g(L)$
Direct derivation $G \Rightarrow_{r,g} H$

Given $r: \langle L \supseteq K \subseteq R \rangle$ and injective morphism $g: L \rightarrow G$

1. Check *dangling condition*: nodes in $g(L - K)$ are not incident to edges outside $g(L)$
2. Remove $g(L - K)$
Direct derivation $G \Rightarrow_{r,g} H$

Given $r: \langle L \supseteq K \subseteq R \rangle$ and injective morphism $g: L \rightarrow G$

1. Check **dangling condition**: nodes in $g(L - K)$ are not incident to edges outside $g(L)$
2. Remove $g(L - K)$
3. Add $R - K$
Hypergraph-transformation systems

Hypergraph-transformation system $\langle \Sigma, \mathcal{R} \rangle$
- $\Sigma$: signature
- $\mathcal{R}$: set of rules over $\Sigma$

Graph-transformation system $\langle \Sigma, \mathcal{R} \rangle$
- for each label $l \in \Sigma_E$ and each $w \in \text{Type}(l)$, $|w| = 2$
Example: reduction rules for control-flow graphs

seq:

while:

dec1:

dec2:
A hypergraph-transformation system \( \langle \Sigma, \mathcal{R} \rangle \) is \textit{confluent} if for all hypergraphs \( G, H_1, H_2 \) over \( \Sigma \), \( H_1 \xleftarrow{\mathcal{R}}^* G \xrightarrow{\mathcal{R}}^* H_2 \) implies that there is a hypergraph \( M \) such that \( H_1 \xrightarrow{\mathcal{R}}^* M \xleftarrow{\mathcal{R}}^* H_2 \).
Confluence

A hypergraph-transformation system \( \langle \Sigma, \mathcal{R} \rangle \) is **confluent** if for all hypergraphs \( G, H_1, H_2 \) over \( \Sigma \), \( H_1 \leftarrow^* \mathcal{R} G \rightarrow^* \mathcal{R} H_2 \) implies that there is a hypergraph \( M \) such that \( H_1 \rightarrow^* \mathcal{R} M \leftarrow^* \mathcal{R} H_2 \).

\[
\begin{array}{c}
G \\
\leftarrow^* \mathcal{R} \\
H_1 \\
\rightarrow^* \mathcal{R} \\
M \\
\rightarrow^* \mathcal{R} \\
H_2 \\
\leftarrow^* \mathcal{R}
\end{array}
\]

*Note*: Confluence implies that every hypergraph can be reduced to at most one irreducible hypergraph (up to isomorphism). The converse does not hold.
Local confluence

A hypergraph-transformation system $\langle \Sigma, \mathcal{R} \rangle$ is \textit{locally confluent} if for all hypergraphs $G, H_1, H_2$ over $\Sigma$, $H_1 \xleftarrow{\mathcal{R}} G \xrightarrow{\mathcal{R}} H_2$ implies that there is a hypergraph $M$ such that $H_1 \xrightarrow{\mathcal{R}}^* M \xleftarrow{\mathcal{R}}^* H_2$. 
Local confluence does not imply confluence

Counterexample:

\[ \mathcal{R} \left\{ \begin{array}{l}
\mathsf{r}_1 : \mathsf{a} \Rightarrow \mathsf{b} \\
\mathsf{r}_2 : \mathsf{a} \Rightarrow \mathsf{c} \\
\mathsf{r}_3 : \mathsf{c} \Rightarrow \mathsf{a} \\
\mathsf{r}_4 : \mathsf{c} \Rightarrow \mathsf{d} \\
\end{array} \right. \]

The system is locally confluent but not confluent:
Termination and confluence

A hypergraph-transformation system $\langle \Sigma, \mathcal{R} \rangle$ is *terminating* if there is no infinite sequence $G_0 \Rightarrow_{\mathcal{R}} G_1 \Rightarrow_{\mathcal{R}} G_2 \Rightarrow_{\mathcal{R}} \ldots$

**Theorem (Newman’s lemma)**

A terminating hypergraph-transformation system is confluent if and only if it is locally confluent.
Critical pairs (for injective rules)

Direct derivations $T_1 \leftarrow_{r_1,g_1} S \Rightarrow_{r_2,g_2} T_2$ are a critical pair if

1. $S = g_1(L_1) \cup g_2(L_2)$ and
2. the steps are not independent, that is, $S \Rightarrow_{r_1,g_1} T_1$ removes an item in $g_2(L_2)$ or $S \Rightarrow_{r_2,g_2} T_2$ removes an item in $g_1(L_1)$.

Also, if $r_1 = r_2$ then $g_1 \neq g_2$ must hold.
Critical pairs (for injective rules)

Direct derivations $T_1 \leftarrow_{r_1,g_1} S \Rightarrow_{r_2,g_2} T_2$ are a critical pair if

1. $S = g_1(L_1) \cup g_2(L_2)$ and
2. the steps are not independent, that is, $S \Rightarrow_{r_1,g_1} T_1$ removes an item in $g_2(L_2)$ or $S \Rightarrow_{r_2,g_2} T_2$ removes an item in $g_1(L_1)$.

Also, if $r_1 = r_2$ then $g_1 \neq g_2$ must hold.

Note: Hypergraph-transformation systems with finitely many rules possess only finitely many critical pairs (up to isomorphism).
Example: critical pairs of control-flow reduction rules (1)
Example: critical pairs of control-flow reduction rules (2)
Example: critical pairs of control-flow reduction rules (3)
Example: critical pairs of control-flow reduction rules (4)

\[
\begin{align*}
\text{seq} & \quad \text{dec2} \\
\text{seq} & \quad \text{dec2}
\end{align*}
\]
Joinability

Given a hypergraph-transformation system $\langle \Sigma, \mathcal{R} \rangle$, a critical pair $T_1 \leftarrow S \rightarrow T_2$ is \textit{joinable} if there is a hypergraph $U$ such that $T_1 \Rightarrow_\mathcal{R}^* U \Leftarrow_\mathcal{R}^* T_2$. 

\[
\begin{array}{c}
S \\
\nearrow \nearrow \\
T_1 \quad \quad \quad \quad \spacebox{60}{T_2} \\
\searrow \searrow \\
U
\end{array}
\]
Joinability is insufficient

Joinability of all critical pairs does not guarantee local confluence:

\[ r_1 : \begin{array}{c}
\text{a} \\
1 \\
2
\end{array} \xrightarrow{} \begin{array}{c}
b \\
1 \\
2
\end{array} \quad \Rightarrow \quad \begin{array}{c}
b \\
1 \\
2
\end{array} \xleftarrow{r_1} \begin{array}{c}
a \\
1 \\
2
\end{array} \xrightarrow{} \begin{array}{c}
b \\
1 \\
2
\end{array} \quad r_2 : \begin{array}{c}
a \\
1 \\
2
\end{array} \xrightarrow{} \begin{array}{c}
1 \\
2
\end{array} \xleftarrow{r_2} \begin{array}{c}
b \\
1 \\
2
\end{array} \]

The only critical pair is joinable:

\[ \begin{array}{c}
b \\
1 \\
2
\end{array} \xleftarrow{r_1} \begin{array}{c}
a \\
1 \\
2
\end{array} \xrightarrow{} \begin{array}{c}
b \\
1 \\
2
\end{array} \xrightarrow{r_2} \begin{array}{c}
1 \\
2
\end{array} \]

But \( \langle \Sigma, \{r_1, r_2\} \rangle \) is not locally confluent:

\[ \begin{array}{c}
b \\
1 \\
2
\end{array} \xleftarrow{r_1} \begin{array}{c}
a \\
1 \\
2
\end{array} \xRightarrow{r_2} \begin{array}{c}
b \\
1 \\
2
\end{array} \]
Strong joinability

For a critical pair $\Gamma : T_1 \Leftarrow S \Rightarrow T_2$, define

$$\text{Persist}(\Gamma) = \{ v \in S_V \mid S \Rightarrow T_1 \text{ and } S \Rightarrow T_2 \text{ preserve } v \}.$$ 

Given a hypergraph-transformation system $\langle \Sigma, \mathcal{R} \rangle$, a critical pair $\Gamma : T_1 \Leftarrow S \Rightarrow T_2$ is strongly joinable if there are a hypergraph $U$ and derivations $T_1 \Rightarrow^*_\mathcal{R} U \Leftarrow^*_\mathcal{R} T_2$ such that for each $v \in \text{Persist}(\Gamma)$, $\text{tr}_{S \Rightarrow T_1 \Rightarrow^* U}(v)$ and $\text{tr}_{S \Rightarrow T_2 \Rightarrow^* U}(v)$ are defined and equal.

Here $\text{tr}_{S \Rightarrow T_1 \Rightarrow^* U} : S \rightarrow U$ and $\text{tr}_{S \Rightarrow T_2 \Rightarrow^* U} : S \rightarrow U$ are partial morphisms that, informally, track the items in $S$ through the derivations $S \Rightarrow T_1 \Rightarrow^* U$ and $S \Rightarrow T_2 \Rightarrow^* U$. 
Critical pair lemma

Theorem (Critical pair lemma [P 93])
A hypergraph-transformation system is locally confluent if all its critical pairs are strongly joinable.

Corollary
A terminating hypergraph-transformation system is confluent if all its critical pairs are strongly joinable.
Confluence does not imply strong joinability

The system

\[ r_1 : \quad \begin{array}{c}
    \rightarrow \\
    1 \\
    \end{array} \rightarrow \\
\begin{array}{c}
    \rightarrow \\
    2 \\
    \end{array} \quad \Rightarrow \quad \begin{array}{c}
    \rightarrow \\
    1 \\
    \end{array} \rightarrow \\
\begin{array}{c}
    \rightarrow \\
    2 \\
    \end{array} \]

\[ r_2 : \quad \begin{array}{c}
    \rightarrow \\
    1 \\
    \end{array} \rightarrow \\
\begin{array}{c}
    \rightarrow \\
    2 \\
    \end{array} \quad \Rightarrow \quad \begin{array}{c}
    \rightarrow \\
    1 \\
    \end{array} \rightarrow \\
\begin{array}{c}
    \rightarrow \\
    2 \\
    \end{array} \]

is terminating and confluent but the critical pair

\[ \begin{array}{c}
    \rightarrow \\
    1 \\
    \end{array} \quad \begin{array}{c}
    \rightarrow \\
    2 \\
    \end{array} \quad \Leftarrow \quad \begin{array}{c}
    \rightarrow \\
    1 \\
    \end{array} \rightarrow \\
\begin{array}{c}
    \rightarrow \\
    2 \\
    \end{array} \quad \Rightarrow \quad \begin{array}{c}
    \rightarrow \\
    1 \\
    \end{array} \rightarrow \\
\begin{array}{c}
    \rightarrow \\
    2 \\
    \end{array} \]

is not strongly joinable
Bad news: confluence is undecidable

**Theorem ([P 93/05])**

The following problem is undecidable in general:

**Instance:** A terminating graph transformation system $\langle \Sigma, R \rangle$ where $\Sigma_V$ is a singleton and $\Sigma_E$ and $R$ are finite.

**Question:** Is $\langle \Sigma, R \rangle$ confluent?
Subhypergraphs $G^\mathcal{R}$ and $G^\ominus$

Let $\langle \Sigma, \mathcal{R} \rangle$ be a hypergraph-transformation system and $G$ a hypergraph over $\Sigma$.

- For $e \in E_G$, the pair $\langle l_E(e), l^*_V(\text{att}_G(e)) \rangle$ is the profile of $e$.
- $\text{Prof}(\mathcal{R})$ is the set of all hyperedge profiles occurring in $\mathcal{R}$.
- $\text{VL}(\mathcal{R})$ is the set of all vertex labels occurring in $\mathcal{R}$.

Define subhypergraphs $G^\mathcal{R}$ and $G^\ominus$ of $G$ as follows.

- $G^\mathcal{R}$ consists of all hyperedges with profile in $\text{Prof}(\mathcal{R})$ and all nodes with label in $\text{VL}(\mathcal{R})$.
- $G^\ominus$ consists of all hyperedges in $E_G - \text{Prof}(\mathcal{R})$, all attachment nodes of these hyperedges, and all nodes in $V_G - \text{VL}(\mathcal{R})$.

Note that $G = G^\mathcal{R} \cup G^\ominus$, where $G^\mathcal{R}$ and $G^\ominus$ may share some attachment nodes of hyperedges in $G^\ominus$. 
Covering critical pairs

A *cover* for a critical pair $\Gamma$ is a hypergraph $C$ such that

1. $\text{Persist}_\Gamma \subseteq C$,
2. $C^\ominus = C$, and
3. for every homomorphic image $\tilde{C}$ of $C$, there is a unique surjective morphism $C \rightarrow \tilde{C}$.

**Remarks:**

1. (1) and (2) imply that each node in $\text{Persist}_\Gamma$ is incident to some hyperedge in $C$.
2. Intuitively, $C$ uniquely identifies the nodes in $\text{Persist}_\Gamma$ in that for every image $\tilde{C}$ of $C$, each node in $\text{Persist}_\Gamma$ corresponds to a unique node in $\tilde{C}$.
3. By (3), $C$ does not possess nontrivial automorphisms.
Example: covers

Let $\Gamma$ be a critical pair such that $\text{Persist}_\Gamma = \{v_1, \ldots, v_n\}$ and $l_V(v_i) = m_i$.

- If there is $l \in \Sigma_E$ such that $m_1 \ldots m_n \in \text{Type}(l)$ and $\langle l, m_1 \ldots m_n \rangle \notin \text{Prof}(\mathcal{R})$, then the following is a cover for $\Gamma$:

  \begin{array}{c}
  l \\
  \downarrow \quad \cdots \quad \downarrow \\
  m_1 \quad \cdots \quad m_n
  \end{array}

- Alternatively, if there are distinct labels $l_1, \ldots, l_{n-1} \in \Sigma_E$ such that for $i = 1, \ldots, n-1$, $m_im_{i+1} \in \text{Type}(l_i)$ and $\langle l_i, m_im_{i+1} \rangle \notin \text{Prof}(\mathcal{R})$, then the following is a graph cover for $\Gamma$:

  \begin{tikzpicture}
  \node[shape=circle] (m1) at (0,0) {$m_1$};
  \node[shape=circle] (l1) at (1,0) {$l_1$};
  \node[shape=circle] (m2) at (2,0) {$m_2$};
  \node[shape=circle] (l2) at (3,0) {$l_2$};
  \node[shape=circle] (mn) at (4,0) {$m_n$};
  \node[shape=circle] (ln) at (5,0) {$l_{n-1}$};
  \draw (m1) -- (l1) -- (m2) -- (l2) -- (mn) -- (ln);
  \end{tikzpicture}
Coverable systems

A hypergraph-transformation system is *coverable* if for each of its critical pairs there exists a cover.

**Theorem**
A coverable hypergraph-transformation system is locally confluent if and only if all its critical pairs are strongly joinable.

Consider now systems $\langle \Sigma, R \rangle$ in which $\Sigma_V$, $\Sigma_E$ and $R$ are finite.

**Corollary**

(1) *Confluence is decidable for terminating coverable systems.*

(2) *Confluence is decidable for terminating systems with universal signatures.*
Decision procedure for confluence

Input: a terminating and coverable hypergraph-transformation system $\langle \Sigma, \mathcal{R} \rangle$ and its set of critical pairs CP

for all $\Gamma: U_1 \leftarrow_{r_1,g_1} S \Rightarrow_{r_2,g_2} U_2$ in CP do

  {let $C$ be a cover for $\Gamma$ such that $S \cap C = \text{Persist}_\Gamma$}

  $\hat{S} := S \cup C$

  {for $i = 1, 2$, let $\hat{g}_i$ be the extension of $g_i$ to $\hat{S}$}

  for $i = 1$ to $2$ do
    construct a derivation $\hat{S} \Rightarrow_{r_i,\hat{g}_i} \hat{U}_i \Rightarrow^*_\mathcal{R} X_i$ such that $X_i$ is irreducible
  end for

if $X_1 \not\cong X_2$ then return “non-confluent”

end if

end for

return “confluent”
Example: extended critical pair
Future work

- Extend the checking algorithm to a partial decision procedure for arbitrary terminating systems: *add* a hyperedge label to an input system such that all critical pairs can be covered and run the algorithm as before. If all extended pairs are joinable, then the (original) system is confluent. If a non-joinable extended pair is encountered whose underlying pair is joinable, then the procedure has to give up.

- Implement critical-pair generation and checking.

- How to check confluence for graph programs?!