MIA 2.0: Richer Interface Automata
with Optimistic and Pessimistic Compatibility

Gerald Lüttgen
Software Technologies Research Group
University of Bamberg, Germany

(Joint work with Walter Vogler, University of Augsburg, Germany)
**MIA 2.0: Further Improvements over IOMTS**

- In our IA-inspired optimistic approach – $\text{MIA}_{\text{opt}}$
  - Making our MIA-setting more practical
  - Exploring the limitations of the MIA-setting

- **Disjunctive must-transitions not only for outputs but also for**
  - *Inputs* – more relaxed meaning of input-determinism
  - *Internal action $\tau$* – more natural encoding of disjunction

- **MIA-refinement as a `better´ IOMTS-refinement**
  - Also considering $\tau$-must-transitions on the specification side
  - Permitting *alphabet extension* on the implementation side

- **Alphabet extension is a crux in $\text{MIA}_{\text{opt}}$**
  - Exploring Bauer et al.‘s *pessimistic view of compatibility* – $\text{MIA}_{\text{pes}}$
MIA_{opt}: Weak Transition Relation

• Not done before in the presence of disjunctive transitions
  – To the best of our knowledge

• Weak may-transition relation \( \Rightarrow \) as usual
  – Leading and trailing \( \tau \)-may-transitions

• Weak output-must-transition relation as the least relation s.t.
  – \( p = \varepsilon \Rightarrow \{p\} \)
  – \( p = \varepsilon \Rightarrow P', \ p' \in P' \) and \( p' - \tau \rightarrow P'' \) implies \( p = \varepsilon \Rightarrow (P' \setminus \{p'\}) \cup P'' \)
  – \( p = \varepsilon \Rightarrow P' = \{p_1, ..., p_n\} \) and \( \forall j. \ p_j - \sigma \rightarrow P_j \) implies \( p = \sigma \Rightarrow \bigcup_j P_j \)
  – \( p = \sigma \Rightarrow P', \ p' \in P' \) and \( p' - \tau \rightarrow P'' \) implies \( p = \sigma \Rightarrow (P' \setminus \{p'\}) \cup P'' \)
MIA_{opt}: MIA-refinement

- Refinement relation as before, but in addition
  - Considering \( \tau \)-must-transitions on the specification side
  - Permitting alphabet extensions on the implementation side

- MIA-refinement \( \leq \)

Given MIAs \( P, Q \) s.t. \( I_P \supseteq I_Q \), \( O_P \supseteq O_Q \) and \( p \leq q \)

1. \( q \cdot i \rightarrow Q' \) implies \( \exists P'. p \cdot i \rightarrow P' \) and \( \forall p' \in P'. \exists q' \in Q'. p' \leq q' \)
2. \( q \cdot \tau \rightarrow Q' \) implies \( \exists P'. p = \varepsilon \Rightarrow P' \) and \( \forall p' \in P'. \exists q' \in Q'. p' \leq q' \)
3. \( q \cdot o \rightarrow Q' \) implies \( \exists P'. p = o \Rightarrow P' \) and \( \forall p' \in P'. \exists q' \in Q'. p' \leq q' \)
4. \( p \cdot i \rightarrow P' \) and \( i \notin I_Q \) implies \( \forall p' \in P'. p' \leq q \)
5. \( p \cdot \tau \rightarrow p' \) implies \( \exists q'. q \rightarrow \varepsilon \mapsto q' \) and \( p' \leq q' \)
6. \( p \cdot o \rightarrow p' \) and \( o \in O_Q \) implies \( \exists q'. q \rightarrow o \mapsto q' \) and \( p' \leq q' \)
7. \( p \cdot o \rightarrow p' \) and \( o \notin O_Q \) implies \( \exists q'. q \rightarrow \varepsilon \mapsto q' \) and \( p' \leq q' \)
MIA\textsubscript{opt}: Parallel Composition

- Compatibility as before, but one new transition rule is needed
  - Let action $a$ be an input of $p$ and an output of $q$, or vice versa
  - (Must3) $(p,q) \to \tau \to P' \times Q'$ if $p \to_a P'$ and $q \to_a Q'$

- Compositionality result still holds
  - Provided an alphabet extension does not result in new communications

- Proof requires a subtle lemma for weak must-transitions
  - If $p = a \implies P'$ and $q \to_a Q'$ then $(p,q) = \epsilon \implies R$ for some $R \subseteq P' \times Q'$
  - Observe that $R = P' \times Q'$ does not hold in general

\begin{itemize}
  \item $1 \xrightarrow{\tau} 2$
  \item $2 \xrightarrow{a!} 4$
  \item $3 \xrightarrow{a!} 5$
  \item $0 \xrightarrow{a?}$
\end{itemize}
MIA_{opt}: Disjunction & Conjunction

- Simplifying assumption, for the moment
  - Operands have the same input/output alphabets

- Disjunction \( \lor \)
  - More intuitive encoding, now using disjunctive \( \tau \)- and input-must-transitions
    - \((Must)\) \( p \lor q - \tau \rightarrow \{p, q\} \)
    - \((IMust)\) \( p \lor q - i \rightarrow P' \cup Q' \) if \( p - i \rightarrow P' \) and \( q - i \rightarrow Q' \)
    - Plus the underlying may-transitions
      - \( \lor \) is the least upper bound wrt. \( \leq \), which implies compositionality

- Conjunction \( \land \)
  - As before, where must-\( \tau \)s are treated in the same way as must-outputs
    - \( \land \) is the greatest lower bound wrt. \( \leq \), which implies compositionality
MIA\text{opt}: Dissimilar Alphabets & Conjunction

- Extending alphabets morally means to add `neutral` may-loops
  - Here, \( p \land q \) should intuitively be implementable by \( u \) and \( v \) but not \( w \)
  - No MIA\text{opt} has \( u \) and \( v \) but not \( w \) as implementations, since it cannot have an initial i?-transition because of \( v \)

- In MIA\text{opt} one cannot forbid an input in some state – crux!
The Pessimistic Approach – MIA\textsubscript{pes}

• Advocated by Bauer, Hennicker and co-workers (MIO [BMSH10])
  – *Pessimistic view of compatibility in an MTS-based interface theory*
  – **Our contributions:** (1) conjunction and disjunction operators,
    (2) disjunctive must-transitions, (3) alphabet extension

• Relaxed MIA
  – Defined as our original MIA but **no requirement for input-determinism**
    and **no requirement that may-inputs are must-inputs**

• Standard modal-refinement on relaxed MIA, i.e., $p \preceq q$ implies
  – *Must-input of $q$ matched by $p$*
    • Possibly with **trailing-only** must-\(	au\)s
  – *Must-output or must-\(	au\) of $q$ matched by $p$*
    • Possibly with leading and/or trailing must-\(	au\)s
  – *May-action of $p$ matched by $q$*
    • Possibly with leading and/or trailing may-\(	au\)s
MIA_{pes}: Compatibility & Parallel Composition

• Restrictive form of compatibility
  – If any error state in a parallel composition can be reached via any action (including an input), then this composition is not defined
  – This approach distinguishes between $\bullet \xrightarrow{i?} \bullet _{E}$ and $\bullet \xrightarrow{i?} \not\rightarrow$

• Parallel composition operator defined as before
  – Compositionality of $\leq$ in MIA_{pes} holds, as to be expected

• Crucial trade-off here
  – Classic view of may-inputs (i.e., input allowed or prohibited) is possible for a large class of systems that must not be input-deterministic
  – But pessimistic view of compatibility means that much fewer parallel compositions are permitted, which is bad for component-based design
MIA_{pes}: Disjunction & Conjunction

• First defined on MIAs with the same input/output alphabets
  – Disjunction exactly as in MIA_{opt}
  – Conjunction simpler as in MIA_{opt}
    • Inputs are treated the same way as outputs and $\tau$

• Lifted to MIAs with dissimilar alphabets
  – Via alphabet extension operator $[p]_A$ (for $A$ disjoint to $p$'s alphabet), which adds an $a$-may-loop to every state of $p$, for every $a$ in $A$
  – $[\_]_A$ is monotonic wrt. $\leq$
  – $[\_]_A$ is a homomorphism for conjunction

• Results, as to be expected
  – $\lor$ is least upper bound  
    and  
    $\leq$ is compositional for $\lor$
  – $\land$ is greatest lower bound  
    and  
    $\leq$ is compositional for $\land$
MIA_{pes}: Conjunction Example

- **Conjuncts with dissimilar alphabets**

  \[
  p \quad \begin{array}{c}
  \bullet \\
  \text{a?} \\
  \rightarrow \\
  \bullet \\
  \text{b!} \\
  \rightarrow \\
  \bullet
  \end{array}
  \]

  \[
  q \quad \begin{array}{c}
  \bullet \\
  \text{a?} \\
  \rightarrow \\
  \bullet \\
  \rightarrow \\
  \bullet \\
  \text{b!} \\
  \rightarrow \\
  \bullet \\
  \rightarrow \\
  \bullet
  \end{array}
  \]

- **Alphabet extension of conjunct \( p \)**

  \[
  [p]_{\{c!\}} \quad \begin{array}{c}
  \bullet \\
  \text{a?} \\
  \rightarrow \\
  \bullet \\
  \text{b!} \\
  \rightarrow \\
  \bullet \\
  \text{c!} \\
  \rightarrow \\
  \bullet \\
  \text{c!} \\
  \rightarrow \\
  \bullet \\
  \text{c!} \\
  \rightarrow \\
  \bullet
  \end{array}
  \]

  \[
  [q]_{\{\}} \quad \begin{array}{c}
  \bullet \\
  \text{a?} \\
  \rightarrow \\
  \bullet \\
  \rightarrow \\
  \bullet \\
  \text{b!} \\
  \rightarrow \\
  \bullet \\
  \rightarrow \\
  \bullet \\
  \text{c!} \\
  \rightarrow \\
  \bullet \\
  \rightarrow \\
  \bullet
  \end{array}
  \]

- **Conjunctive composition**

  \[
  p \land q = [p]_{\{c!\}} \land [q]_{\{\}}
  \]

\[
\begin{array}{c}
  \bullet \\
  \text{a?} \\
  \rightarrow \\
  \bullet \\
  \rightarrow \\
  \bullet \\
  \text{c!} \\
  \rightarrow \\
  \bullet
  \end{array}
  \]

\[
\begin{array}{c}
  \bullet \\
  \rightarrow \\
  \bullet \\
  \rightarrow \\
  \bullet
  \end{array}
  \]

\[
\begin{array}{c}
  \bullet \\
  \text{b!} \\
  \rightarrow \\
  \bullet
  \end{array}
  \]
## Summary

### Optimistic Approach
- **IA, IOMTS, MIA\textsubscript{opt}**
- **True open-systems view**
  - Generous compatibility
- **IA-style modal refinement**
  - Inputs: required, allowed
  - Input-determinism
- **Alphabet extension**
  - Conjunction and disjunction restricted to same alphabets

### Pessimistic Approach
- **MIO, MIA\textsubscript{pes}**
- **Blurred open-systems view**
  - Restrictive compatibility
- **Standard modal refinement**
  - Inputs: required, allowed, **forbidden**
  - No input-determinism
- **General alphabet extension**
  - Alphabet extension operator
Conclusions & Future Work

• MIA is a well-founded interface theory that allows one to
  – Enforce outputs
  – *Express disjunctive must-transitions*
  – Specify non-deterministic behaviour
  – Abstract from internal computation
  – Interpret compatibility optimistically or pessimistically
  – *Compose interfaces also conjunctively and disjunctively*
  – *Extend alphabets while refining*

• Some questions left to future work
  – Are there `clean´ interface theories in-between the optimistic and pessimistic approaches?
  – *Is there a possibility to allow may-inputs while maintaining a true open-systems view?*
Selected Literature


