Modal Interface Automata

Gerald Lüttgen
Lehrstuhl Softwaretechnik und Programmiersprachen
Universität Bamberg

(Joint work with Walter Vogler, Universität Augsburg)

Interface Theories

- Traditional interfaces *(from 1960s)*
  - Deal with *types of data and operations*
  - Are implemented in most compilers
- Sequential, object-oriented interfaces *(from mid 1980s)*
  - *Behavioural types*, as supported by automated verification tools
  - E.g., pre-/post-conditions and invariants of classes and methods
  - Prominent example: **Meyer's contracts in Eiffel**
- Concurrent, behavioural interfaces *(from early 2000s)*
  - *Take concurrency into account*
  - Describe the interaction between system components
  - Prominent prototype: **de Alfaro/Henzinger's Interface Automata (IA)**

Semantic Framework

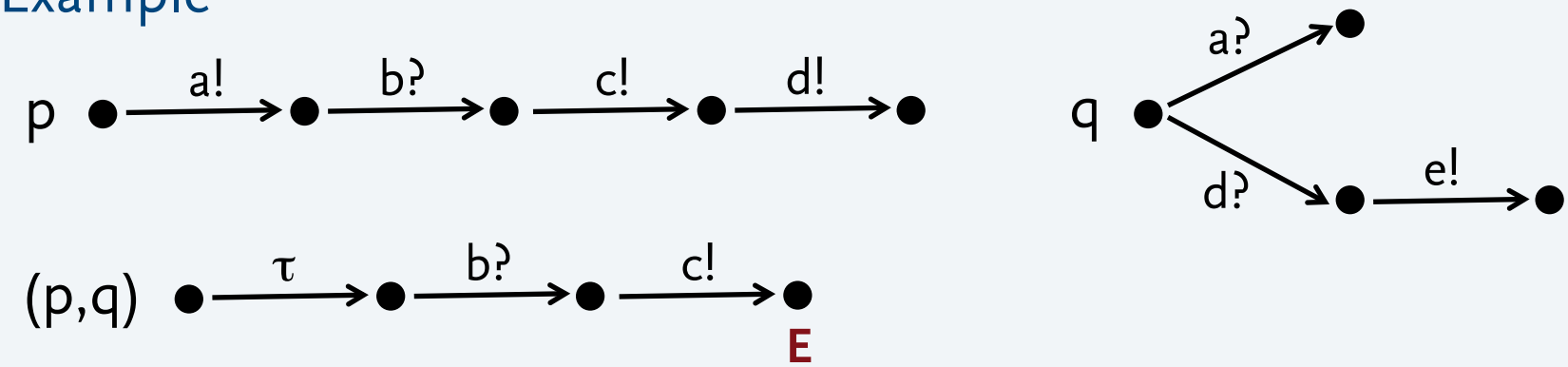
- **Labelled transition system** representing component behaviour
  - *States* p connected via *transitions* p‘ – a –> p‘‘
  - Transition labelled with *input* (i?), *output* (o!) or *internal* (τ) *action*, taken from some alphabet

- Desiderata
  - **Parallel composition**
    - p|q with a sensible notion of *component compatibility*
  - **Compositional refinement preorder**
    - $p \leq q$ implies $p|r \leq q|r$ *(monotonicity of parallel composition)*
  - **Conjunction**
    - Supporting perspective-based specification
    - $r \leq p \wedge q$ if and only if $r \leq p$ and $r \leq q$ *(greatest lower bound)*
  - **Alphabet extension**
    - Permitting the addition of new `features´ when refining interfaces

Parallel Composition on IA

- Basic idea adopted from process algebra *(cf. Milner's CCS)*
  - *Synchronize on shared actions (a!, a?), resulting in the internal action τ*
  - Interleave on non-shared actions

- Example



- Optimistic notion of compatibility [Dill, late 1980s]
  - *Inputs can be delayed, outputs are immediate*
  - If output of p is not expected by q, then p|q is an **error state**
  - Remove states that can reach an error locally (i.e., via outputs or τs)

IA-Refinement

- An alternating simulation: the largest relation $\leq$ on states s. t. for all $p \leq q$
  - $q - a? \rightarrow q'$ implies $\exists\, p'.\ p - a? \rightarrow p'$ and $p' \leq q'$
  - $p - a! \rightarrow p'$ implies $\exists\, q'.\ q\ (- \tau \rightarrow)^* - a! \rightarrow q'$ and $p' \leq q'$

    (and similarly for τ-actions)

- Intuition
  - *Specified inputs* (outputs) *must be implemented* (are not required)
  - *Unspecified inputs* (outputs) *are always allowed* (are forbidden)

- Compositionality result for input-deterministic IA
  - **$p \leq q$ implies $p|r \leq q|r$ for any input-deterministic r**
  - Direct matching of inputs (i.e., without leading τs) is important

Modal Transition Systems (MTS)

- *Interface automata cannot enforce outputs*

≤ p *(for any interface automaton p)*

- Adopt idea of modal transition systems [Larsen, early 1990s]
  - *Add a must-/may-modality to each transition*
    - Must- (may-)transitions must (may) be implemented
    - Unspecified transitions are prohibited in any implementation/refinement
  - *Syntactic consistency, i.e., every must-transition is also a may-transition*

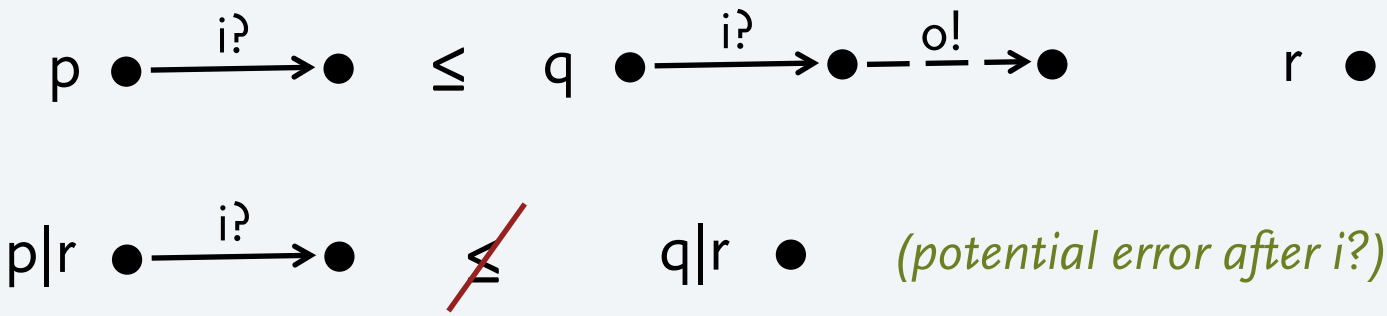
- Modal-refinement is again an alternating simulation ≤ s. t.
  - q – a –> q‘ implies ∃ p‘. p – a –> p‘ and p‘ ≤ q‘
  - p -- a --> p‘ implies ∃ q‘. q (-- τ -->)* -- a --> q‘ and p‘ ≤ q‘

    (and similarly for τ-actions; τ is never must)

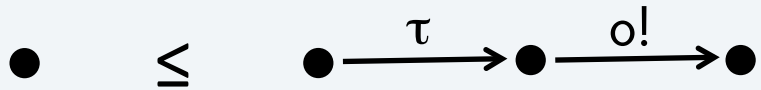
I/O Modal Transition Systems (IOMTS)

- Devised by Larsen, Nyman and Wasowski [ESOP'07]
  - MTS with input and output, as well as classic modal-refinement
  - Parallel composition and compatibility as in IA
  - *Embedding of IA in MTS: inputs (outputs) are must- (may-)transitions*

- **Compositionality flaw** (alphabet of p, q is {i?, o!}; alphabet of r is {o?})

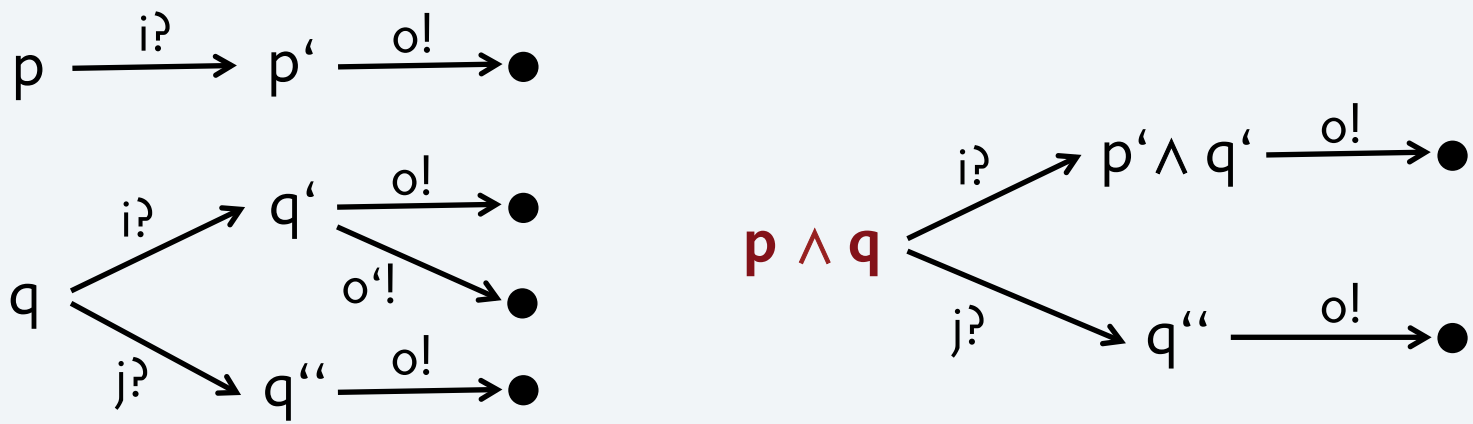


- **No consideration of τ-must-transitions on specification side**



Conjunction on IA

- Example, using interfaces that have the same alphabets

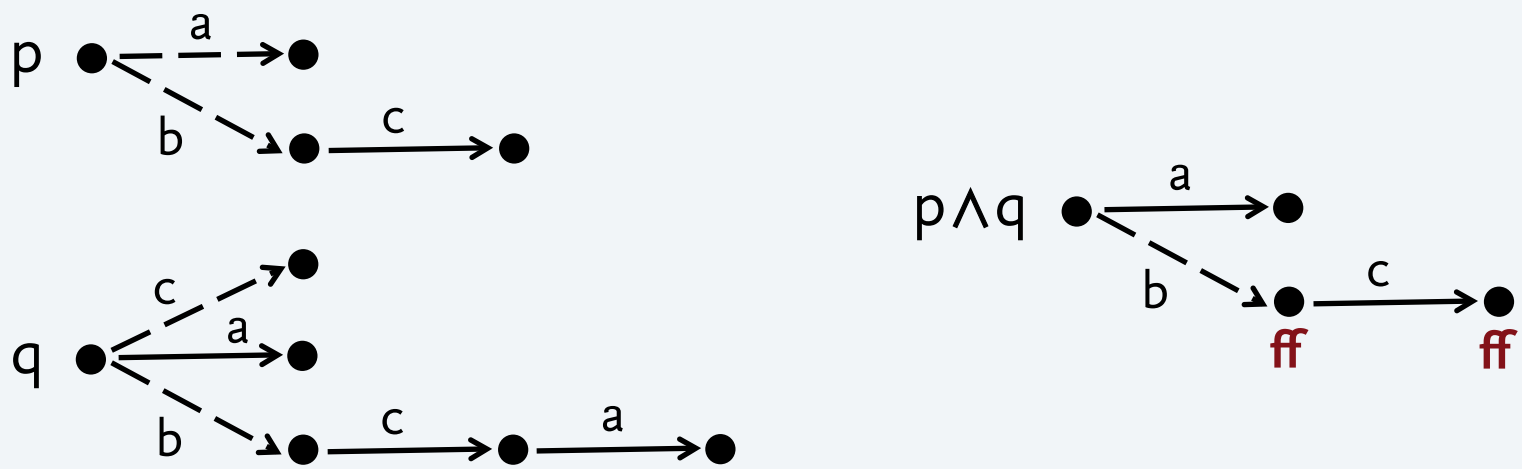


- Any set of interfaces can always be implemented

- *No inconsistencies arise under IA-conjunction*

Conjunction on MTS

- Example (cf. Benes, Cerna and Kretinsky [ATVA'11])



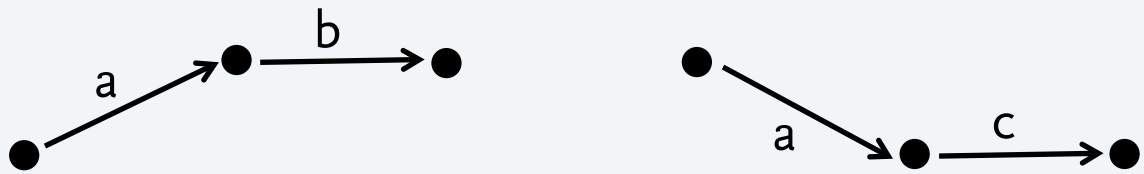
  - After action c, something impossible must be done
  - **Inconsistency propagates backwards**
  - Pruning similar to the one in parallel composition for IA

- Our contribution here relates to dealing with τ-transitions
  - E.g., if $p\ (\text{--}\ \tau\ \text{-->})^+\ p'$ and $q\ (\text{--}\ \tau\ \text{-->})^+\ q'$ then $p \wedge q\ \text{--}\ \tau\ \text{-->}\ p' \wedge q'$
  - **Conjunction is no parallel composition!**

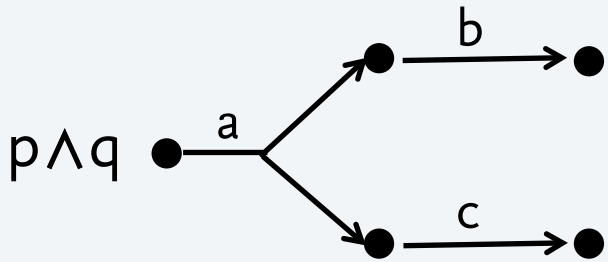
Conjunction: Disjunctive Must-Transitions

- Another example

- Common refinement must have at least one of



- Disjunctive must-transitions



- *Disjunctive must-transitions are easier than syntactic inconsistency*

Modal Interface Automata (MIA)

- Inherits from IA
  - *Parallel composition*, including the notion of compatibility
  - *Input-determinism*, but relaxed due to disjunctive input-must-transitions
- Inherits from MTS
  - *Disjunctive must-transitions* incl. τ-must-transitions, unlike in IOMTS
- Merges ideas from IA and MTS
  - *Refinement* as in MTS, but inputs are always allowed as in IA
    - All may-inputs are must-inputs
    - Parallel composition will be monotonic, unlike in IOMTS
    - Must-τs on specification side will be considered, unlike in IOMTS
  - *Conjunction*, where inputs (outputs) are treated as in IA (MTS)
- Investigates/considers alphabet extension
  - As advocated by Raclet, Caillaud et al. [Fund. Inform., 2011]

Weak Transition Relation

- *Not done before in the presence of disjunctive transitions*
  - To the best of our knowledge
- Weak may-transition relation $\Rrightarrow$ as usual
  - Leading and trailing τ-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 $p = \varepsilon \Rightarrow (P' \setminus \{p'\}) \cup P''$
  - **$p = \varepsilon \Rightarrow P' = \{p_1, \ldots, p_n\}$ and $\forall j.\ p_j - o \rightarrow P_j$ implies $p = o \Rightarrow \bigcup_j P_j$**
  - $p = o \Rightarrow P'$, $p' \in P'$ and $p' - \tau \rightarrow P''$ implies $p = o \Rightarrow (P' \setminus \{p'\}) \cup P''$

MIA-Refinement

- Refinement preorder
  - **Considers τ-must-transitions on the specification side**

- MIA-refinement ≤

  Given MIAs P, Q with the same input and output alphabets and p≤q:

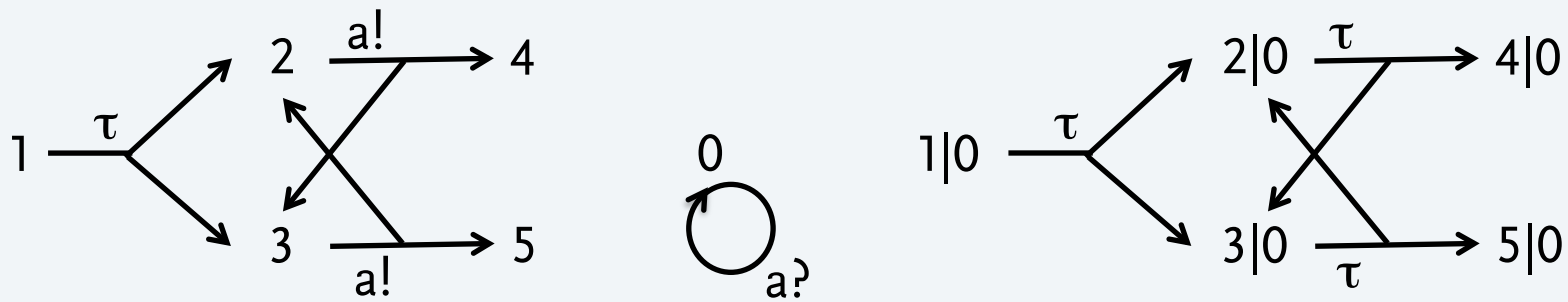
  1. $q - i \rightarrow Q'$ implies $\exists P'.\ p - i \rightarrow P'$ and $\forall p' \in P'\ \exists q' \in Q'.\ p' \leq q'$
  2. $q - o \rightarrow Q'$ implies $\exists P'.\ p = o \Rightarrow P'$ and $\forall p' \in P'\ \exists q' \in Q'.\ p' \leq q'$
  3. **$q - \tau \rightarrow Q'$ implies $\exists P'.\ p = \varepsilon \Rightarrow P'$ and $\forall p' \in P'\ \exists q' \in Q'.\ p' \leq q'$**

  4. *No requirement for input transitions, i.e., inputs are always allowed*
  5. $p \cdots o \cdots\rangle\ p'$ implies $\exists q'.\ q \vdots\ o \vdots\rangle\ q'$ and $p' \leq q'$
  6. $p \cdots \tau \cdots\rangle\ p'$ implies $\exists q'.\ q \vdots\ \varepsilon \vdots\rangle\ q'$ and $p' \leq q'$

Parallel Composition on MIA

- Compatibility as before, but a new transition rule is needed
  - *Let action a be an input of p and an output of q, or vice versa*
  - p|q – τ –> P'|Q' if p – a –> P' and q – a –> Q' *(and symmetrically)*

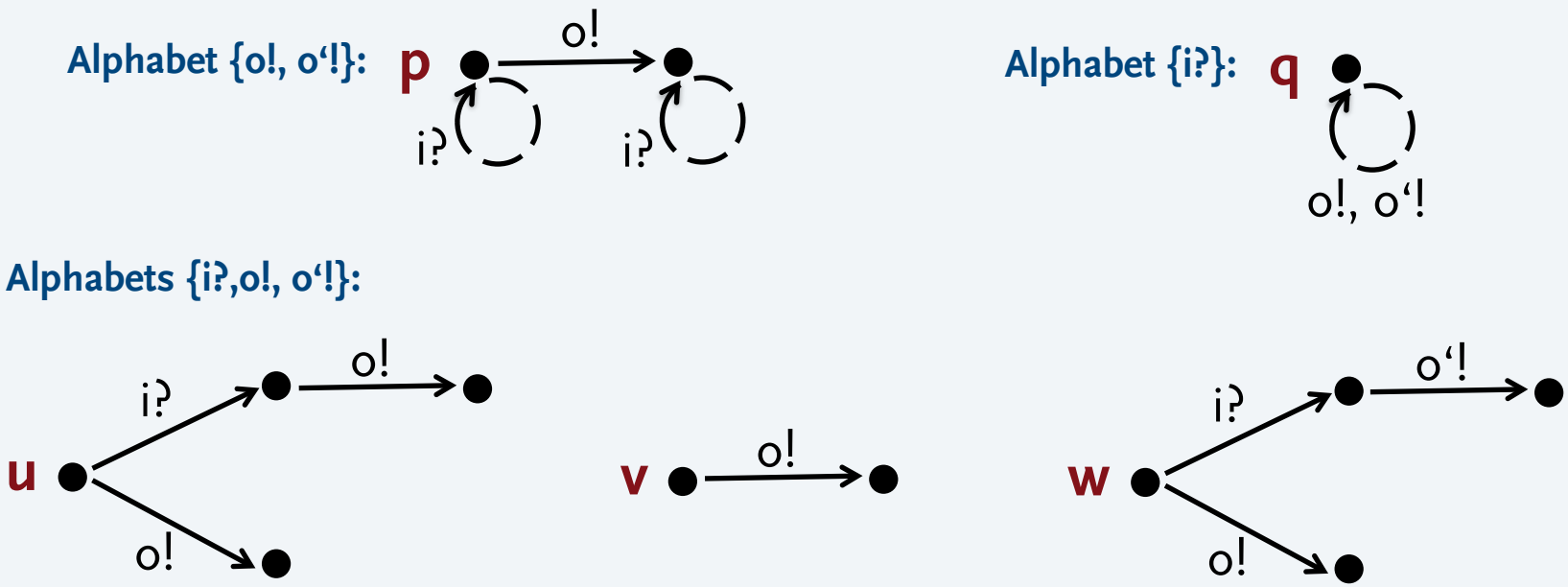
- Compositionality result still holds, but its proof requires a subtle lemma for weak must-transitions
  - If p = a => P' and q – a –> Q' then p|q = ε => R for some R⊆P'|Q'
  - *Observe that R = P'|Q' does not hold in general*



Disjunction & Conjunction on MIA

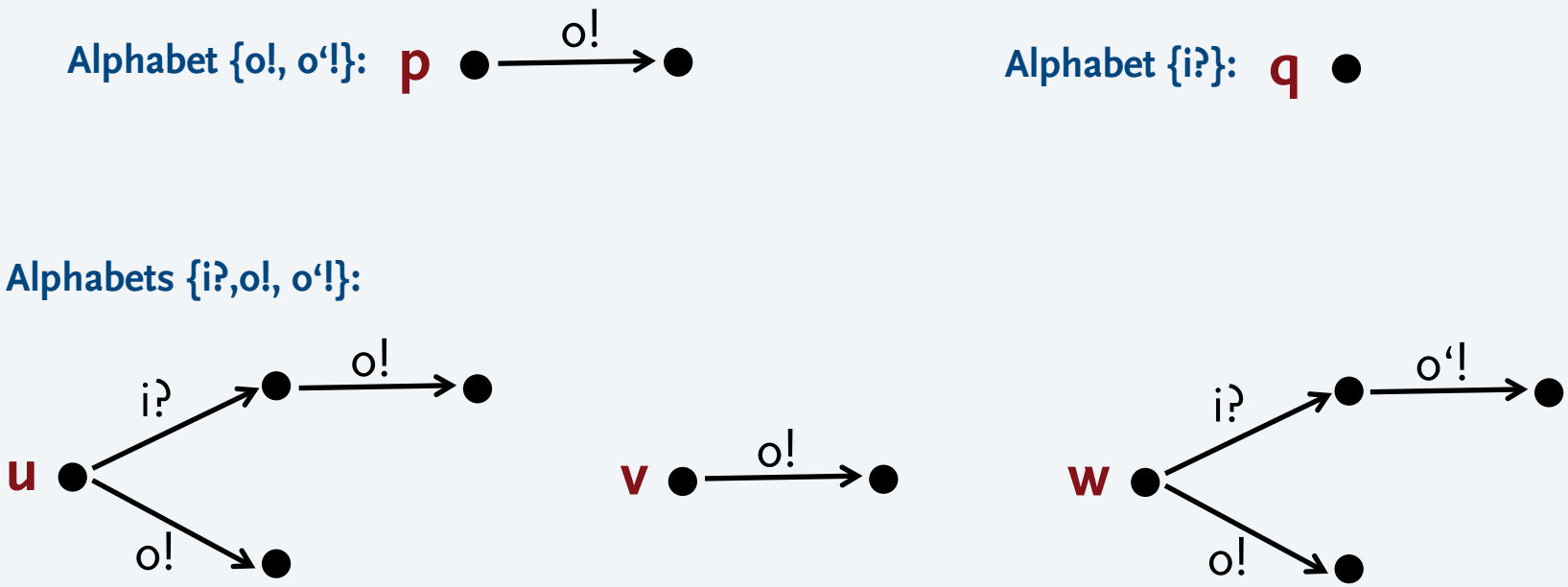
- Disjunction
  - *Intuitive encoding using disjunctive τ-must- and input-must-transitions*
    - $p \vee q - \tau \rightarrow \{p, q\}$
    - $p \vee q - i \rightarrow P' \cup Q'$ if $p - i \rightarrow P'$ and $q - i \rightarrow Q'$
    - Plus the underlying may-transitions
  - $\vee$ is the least upper bound wrt. $\leq$, which implies compositionality
- Conjunction
  - Defined – for the moment – on interfaces with the same alphabets
  - **Treatment of inputs as for IA and of outputs and τs as for MTS**
  - $\wedge$ is the greatest lower bound wrt. $\leq$, which implies compositionality

Conjunction Problem for Dissimilar Alphabets



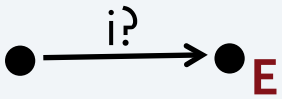
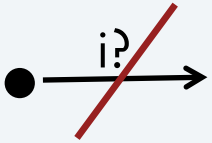
- *Extending alphabets morally means adding `neutral´ may-loops*
  - Here, **p∧q** should intuitively be implementable by **u** and **v** but not **w**
  - **No MIA r** has **u** and **v** but not **w** as implementations, since if **v** refines **r** then also **w** must refine **r**
  - In MIA one may always add some input-transition with arbitrary subsequent behaviour

Alphabet Extension Problem



- *Generalizing refinement ≤ so that the refining interface may have new inputs i is problematic*
  - Consider demanding for <u>new</u> i: **w – i –> W' implies ∀w'∈W'. w'≤p**
  - **Then, refinement would not be transitive since w ≤ v ≤ p but <u>not</u> w ≤ p**

The Pessimistic Approach

- Restrictive, pessimistic notion of compatibility
  - Advocated by Bauer, Hennicker et al. in their MIO theory [TACAS'10]
  - *If any error state in a parallel composition can be reached via **any** actions (including input actions), then the composition is not defined*
  - This approach distinguishes between  and 

- Trade-off
  - *Classic view of may-inputs* (i.e., input allowed or prohibited)
  - **Much fewer parallel compositions are permitted**
- Investigating MIA for pessimistic compatibility
  - *Here, one can do away with input-determinism and with the requirement that may-inputs are also must-inputs*
  - `Standard´ modal-refinement, with compositional parallel operator

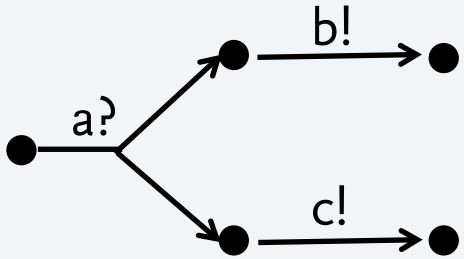
Conjunction with Alphabet Extension

- *Conjuncts with dissimilar alphabets*

- *Alphabet extension* **$[p]_{\{c!\}}$** *of conjunct p*

- *Conjunctive composition*

$$p \wedge q = [p]_{\{c!\}} \wedge [q]_{\{\}}$$



Summary & Comparison

Optimistic Approach

- IA, IOMTS, MIA_{opt}
- **True open-systems view**
  - Generous compatibility
- *IA-style modal refinement*
  - Inputs: required, allowed
  - Input-determinism
- *No alphabet extension capturing perspective-based specification*
  - **Extension is problematic**
  - Logic operators restricted to MIAs with same alphabets

Pessimistic Approach

- MIO, MIA_{pes}
- **Blurred open-systems view**
  - Restrictive compatibility
- *Standard modal refinement*
  - Inputs: required, allowed, **forbidden**
  - No input-determinism needed
- *Alphabet extension capturing perspective-based specification*
  - Alphabet extension operator

Conclusions & Future Work

- **MIA is a compositional interface theory that allows one to**
  - Enforce outputs
  - Express disjunctive must-transitions
  - Specify non-deterministic behaviour
  - Abstract from internal computation
  - Compose interfaces in parallel, conjunctively and disjunctively
  - Interpret compatibility optimistically or pessimistically
  - Extend alphabets while refining (in the pessimistic version)

- Questions left for future work
  - Are there `clean´ interface theories in-between the optimistic and pessimistic approaches?
  - *Is there a possibility to allow new may-inputs while maintaining a true open-systems view?*

Selected Literature

- L. de Alfaro, T. Henzinger. *Interface-based design*. In Engineering Theories of Software-Intensive Systems. NATO Science Series 195. Springer, 2005.
- S. Bauer, P. Mayer, A. Schroeder, R. Hennicker. *On weak modal compatibility, refinement, and the MIO Workbench*. In TACAS'10, LNCS 6015, pp. 175-189. Springer, 2010.
- K. Larsen, U. Nyman, A. Wasowski. *Modal I/O automata for interface and product line theories*. In ESOP'07. LNCS 4421, pp. 64-79. Springer, 2007.
- G. Lüttgen, W. Vogler. *Modal Interface Automata*. LMCS 9(3:4), August 2013.
- G. Lüttgen, W. Vogler. *Richer interface automata with optimistic & pessimistic compatibility*. In AVoCS'13. ECEASST 66, December 2013.
- J. Raclet, E. Badouel, A. Benveniste, B. Caillaud, A. Legay, R. Passerone. *A modal interface theory for component-based design*. Fund. Inform. 107:1-32, 2011.