Branching pomsets and event structures

Luc Edixhoven[1,2] Sung-Shik Jongmans[1]

José Proença[3] Ilaria Castellani[4]

[1]Open University of the Netherlands [2]LIACS, Leiden University

[3]CISTER, ISEP, Polytechnic Institute of Porto [4]INRIA, Université Côte d'Azur

RaPS 2024

Branching pomsets for choreographies

Luc Edixhoven[1,2] Guillermina Cledou[3,4]

Sung-Shik Jongmans[1,2] José Proença[5]

[1] Open University of the Netherlands [2] CWI

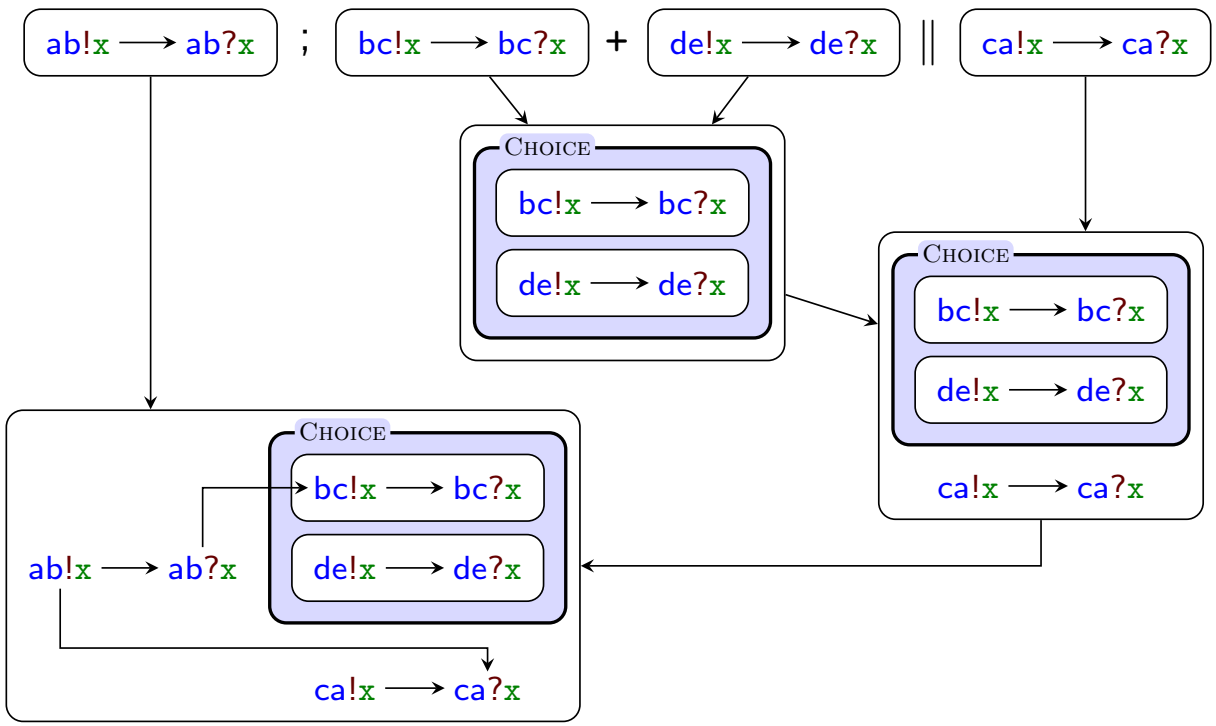
[3] HASLab, INESC TEC [4] University of Minho

[5] CISTER, ISEP, Polytechnic Institute of Porto

ICE 2022

Branching pomsets for choreographies

$$[\![a \to b{:}x \;;\; ((b \to c{:}x + d \to e{:}x) \parallel c \to a{:}x)]\!]$$



21 / 23

Conclusions and future work

Summary

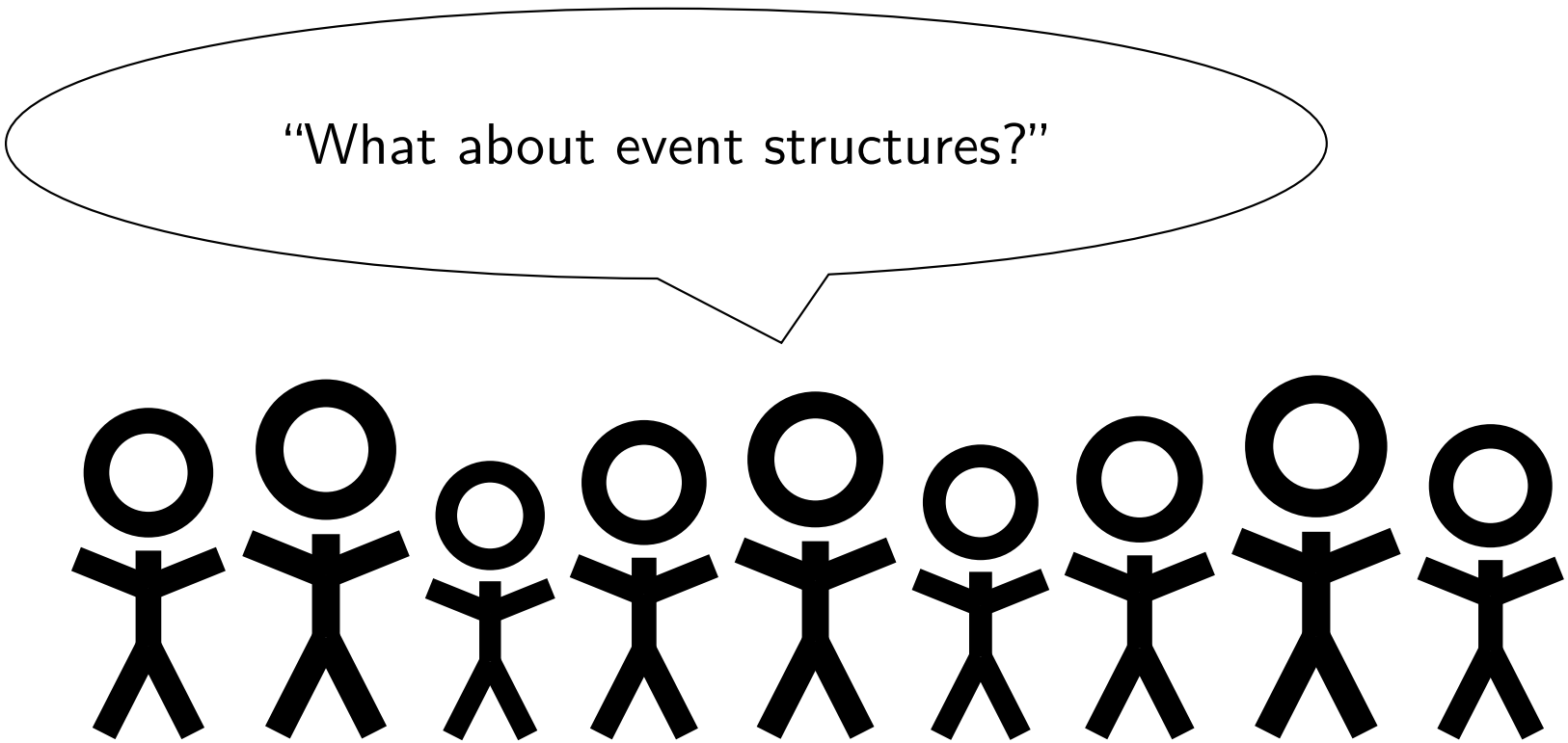
- Branching pomsets
- Compact for both concurrency and choice
- Can express the same behaviour as choreographies

Future work

- Framework improvements: n-ary choices, partial order, loops
- Static analysis: realisability

`https://lmf.di.uminho.pt/b-pomset/`

Some context



Some context

Branching pomsets and event structures (oral communication)

Luc Edixhoven[1,2] Sung-Shik Jongmans[1,2]

José Proença[3] Ilaria Castellani[4]

[1]Open University of the Netherlands [2]CWI

[3]CISTER, ISEP, Polytechnic Institute of Porto [4]INRIA, Université Côte d'Azur

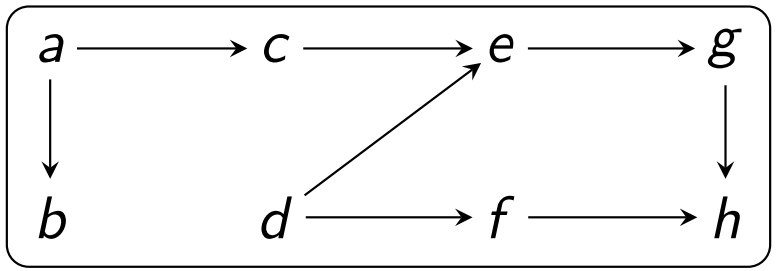
ICE 2023

Outline

- **Branching pomsets**: a generic model for concurrency
- **Event structures**: a brief overview of the landscape
- **Comparison**: relative expressiveness

Branching pomsets and event structures

Basis: partially ordered multisets / pomsets (Pratt 1986)



- a set of events

  above: $\{a, b, c, d, e, f, g, h\}$

- a partial order on the events

  above: the reflexive and transitive closure of the arrows

- a labelling function from events to some set of labels

  above: omitted / identity (irrelevant for this talk)

Branching pomsets and event structures

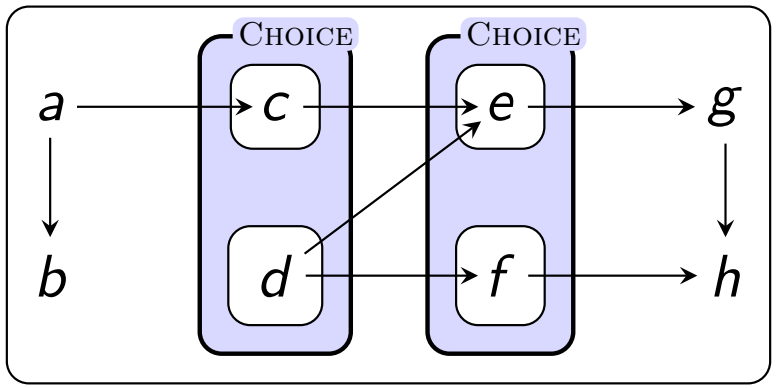
Extension: choices

- expressing choices with pomsets requires a set of pomsets
- with many choices, this set may become exponentially large
- solution: add a representation of choices

Result: a set of pomsets as a single, compact object

Branching pomsets

Choice model: branching structure



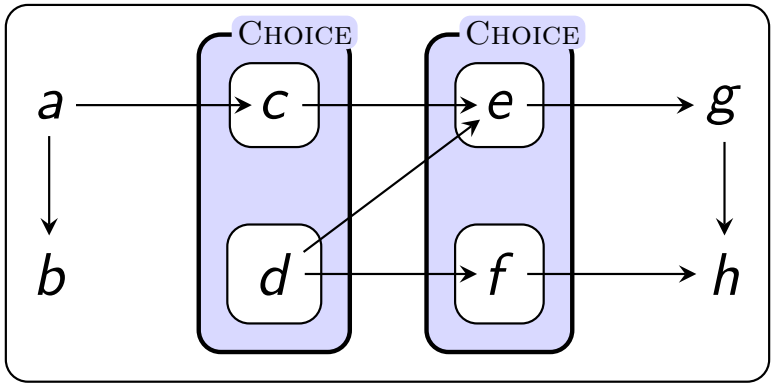
- add branching structure; a tree whose leaves are the events

  above: $\{a, b, g, h, \mathcal{C}_1, \mathcal{C}_2\}$, (visualised as nested boxes)
  where $\mathcal{C}_1 = \{\{c\}, \{d\}\}$ and $\mathcal{C}_2 = \{\{e\}, \{f\}\}$

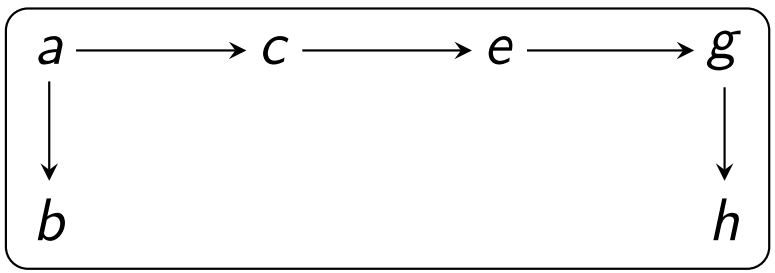
- replace the partial order with a precedence relation, whose reflexive and transitive closure is a partial order

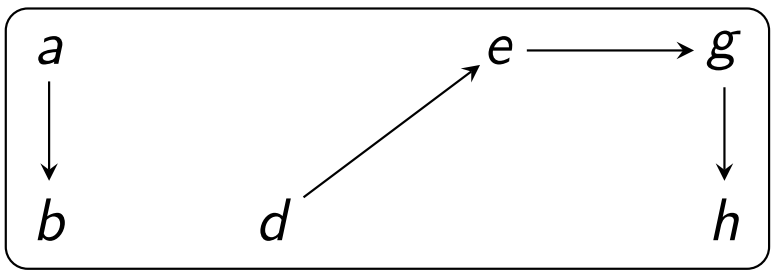
  above: the arrows

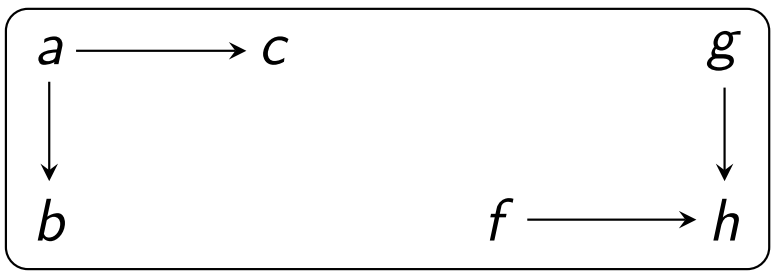
Branching pomsets

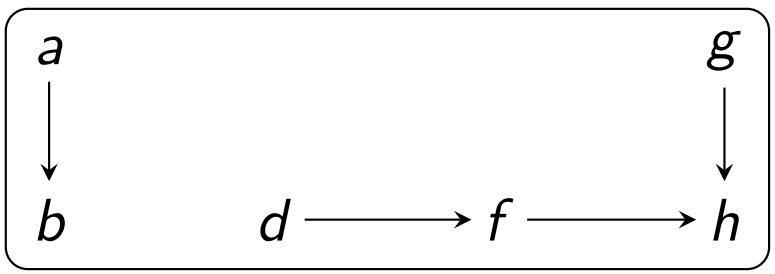


For comparison: the corresponding set of pomsets









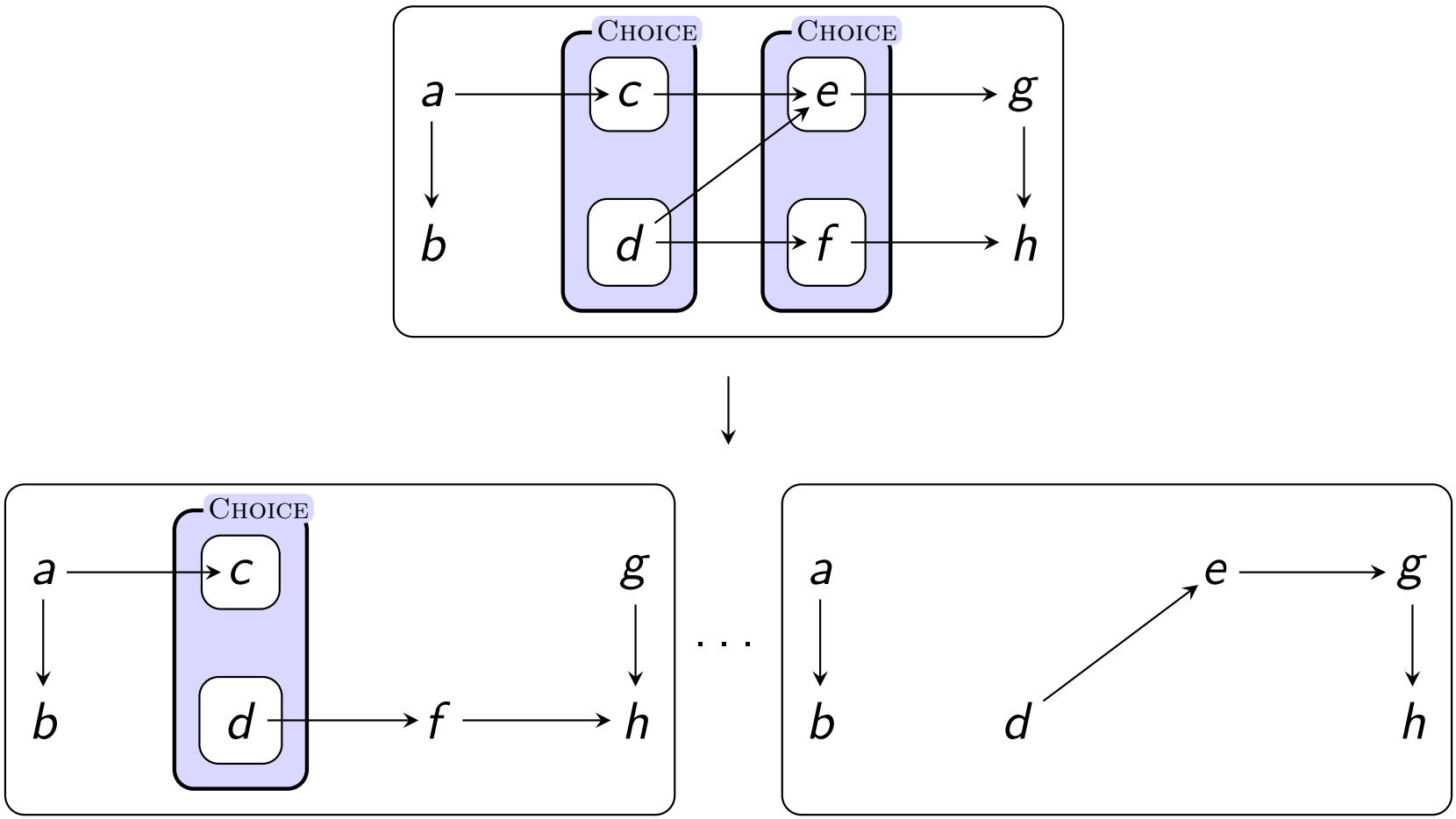
Branching pomsets

Semantics

- should be consistent with that of the corresponding set of pomsets
- formalised using two relations: refining and enabling

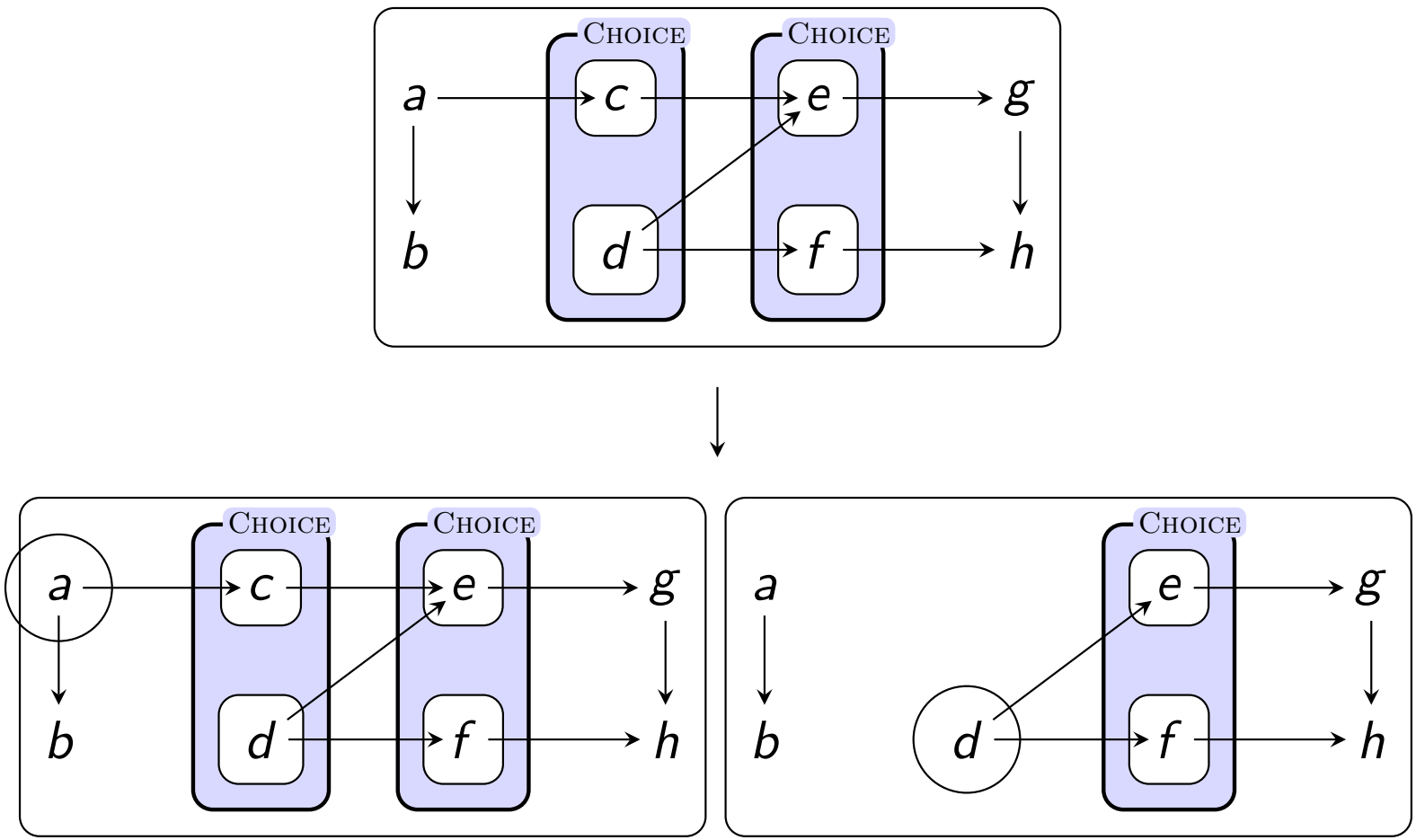
Branching pomsets

Semantics: refining $\Rightarrow$ resolving any number of choices



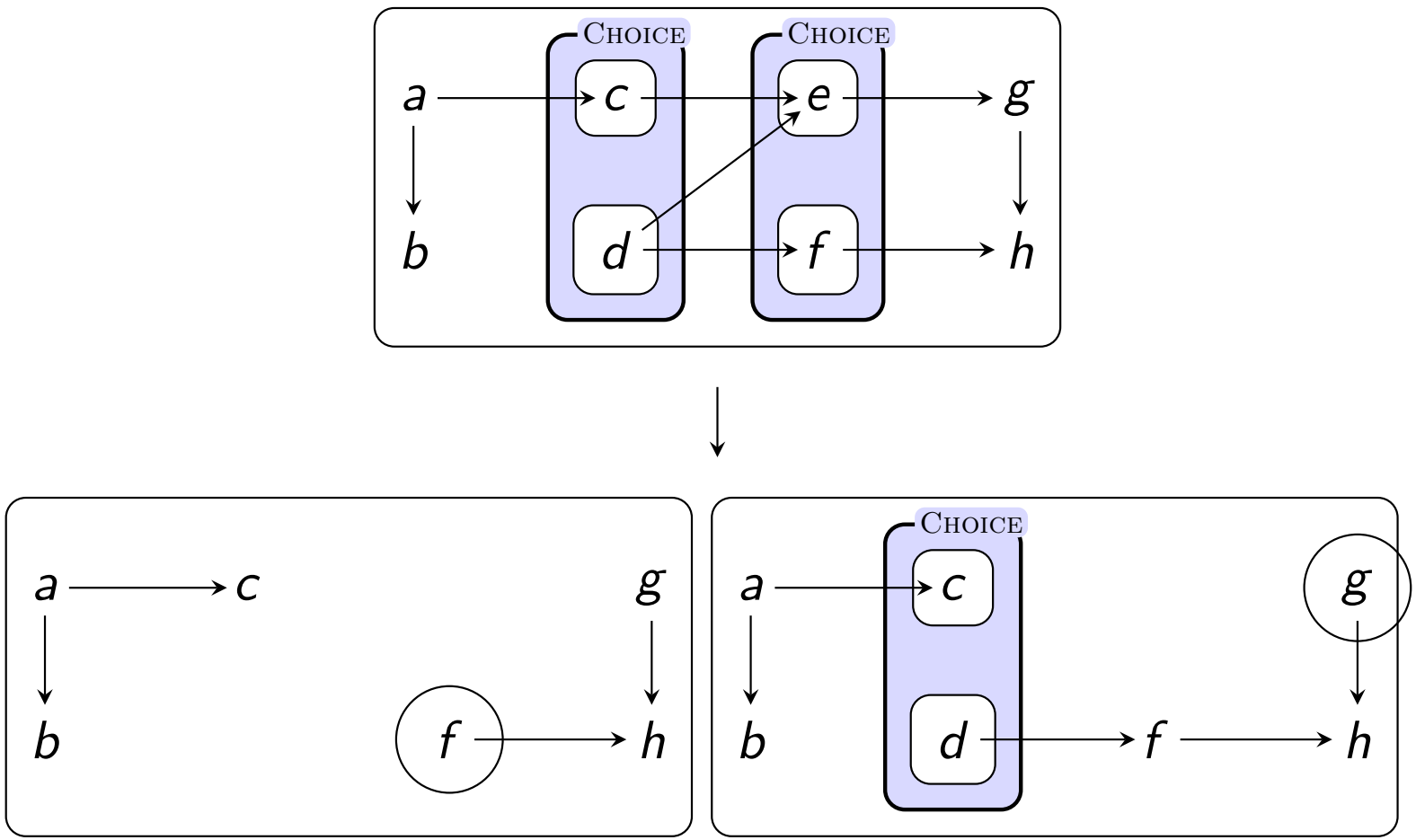
Branching pomsets

Semantics: enabling (followed by firing) $\Rightarrow$ refining s.t. the chosen event is minimal and top-level, resolving no more than necessary



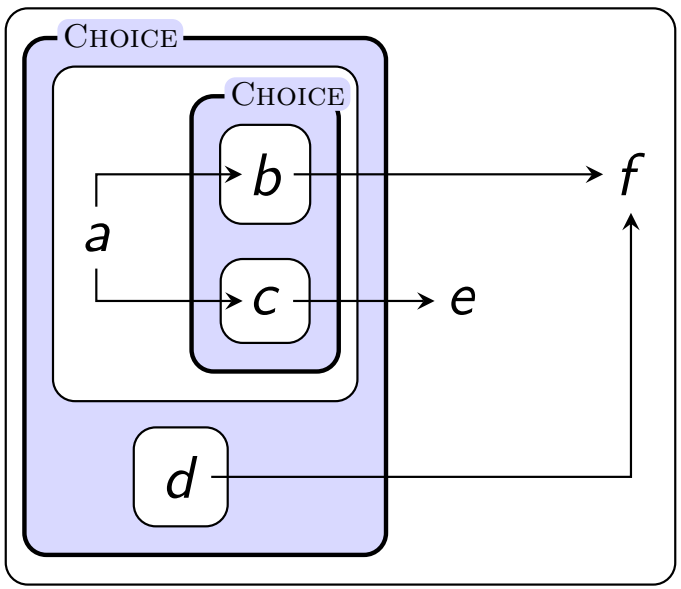
Branching pomsets

Semantics: enabling (followed by firing) $\Rightarrow$ refining s.t. the chosen event is minimal and top-level, resolving no more than necessary



Branching pomsets

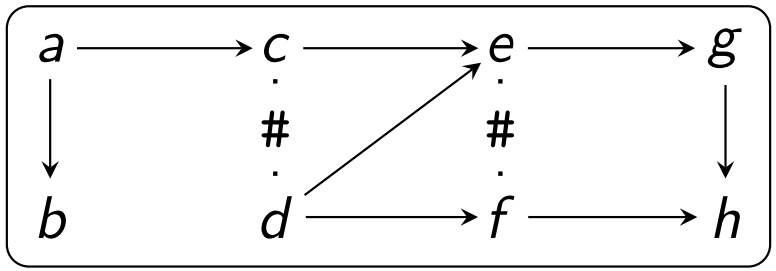
Also: nested choices



Event structures

Nielsen, Plotkin and Winskel (1981)

Choice model: conflict relation



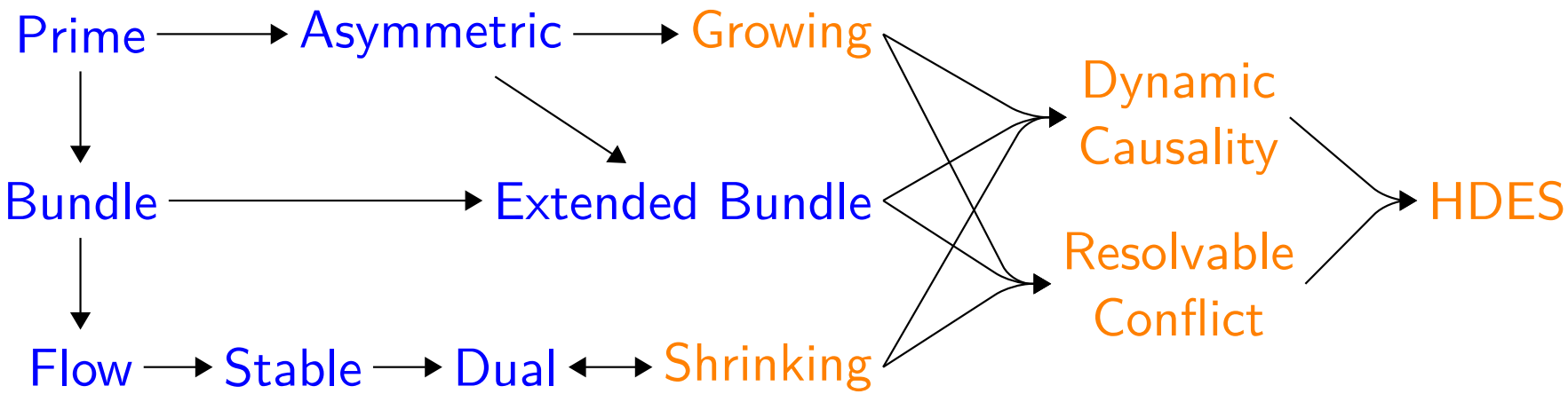
- add conflict relation #; two conflicting events may not occur together in the same execution

  above: $\{(c, d), (e, f)\}$

- most classes of event structures define variations on causality and/or conflicts

Event structures

Landscape (partial): static and dynamic classes of event structures

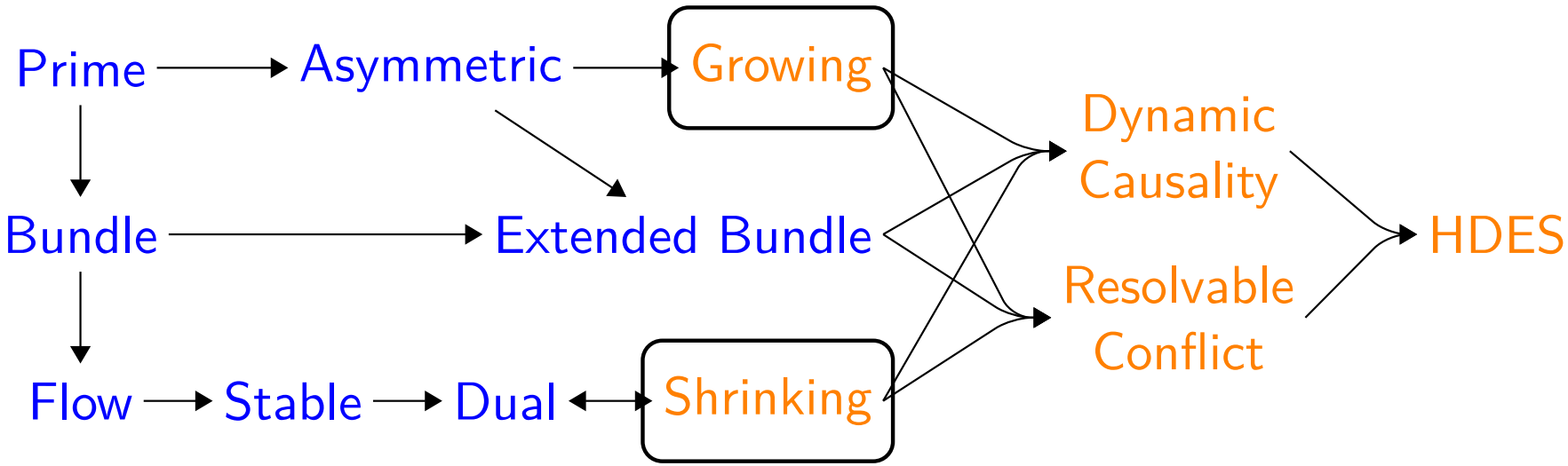


Arrows represent (strict) inclusion in terms of expressiveness

Figure: Arbach, Karcher, Peters and Nestmann, Dynamic causality in event structures (2018)

Event structures

Landscape (partial): static and dynamic classes of event structures



Arrows represent (strict) inclusion in terms of expressiveness

Figure: Arbach, Karcher, Peters and Nestmann, Dynamic causality in event structures (2018)

Most relevant for this talk: growing and shrinking causality $\Rightarrow$ dynamically adding and removing causalities

Event structures

Where it all begins: prime event structures

- partially ordered causality relation
- symmetric conflict relation
- conflict hereditariness: events "inherit" conflicts from their predecessors

Furthermore: an event may not have conflicting causes.

Consequently, limited in expressiveness / compactness. Roughly two main lines of extensions in our part of the landscape.

Event structures

Path: disabling currently enabled events

- (standard) conflict relation $a \# b$: if one of a and b happens, then the other is forbidden
- asymmetric conflict relation $a \rightsquigarrow b$: if a happens first then b may still happen, but if b happens first then a is forbidden.
- growing causality relation $a \blacktriangleright [b \to c]$: a happening will make c causally dependent on b (i.e., add an arrow from b to c)

Event structures

Path: enabling currently disabled events

- (standard) causality relation $a \leq b$: b may not happen before a
- disjunctive causality (bundle, dual) $\{a_1, \ldots, a_n\} \rightarrowtail b$: b may not happen before *some* a_i has happened

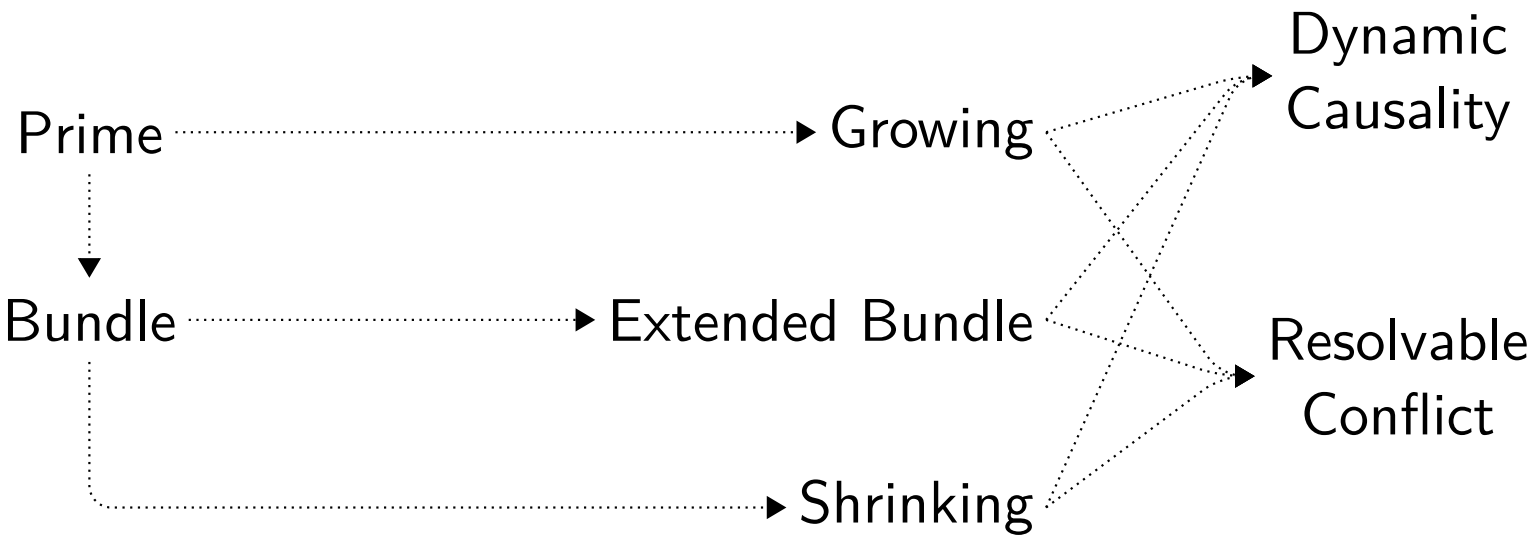
  The a_i may then be required to be in pairwise conflict, depending on the class of event structures.
- enabling relation $\{a_1, \ldots, a_n\} \vdash b$ (stable): b is enabled by the set of events $\{a_1, \ldots, a_n\}$
- shrinking causality relation $a \lhd [b \to c]$: a happening will make c causally independent from b (i.e., remove any arrow from b to c)

Event structures

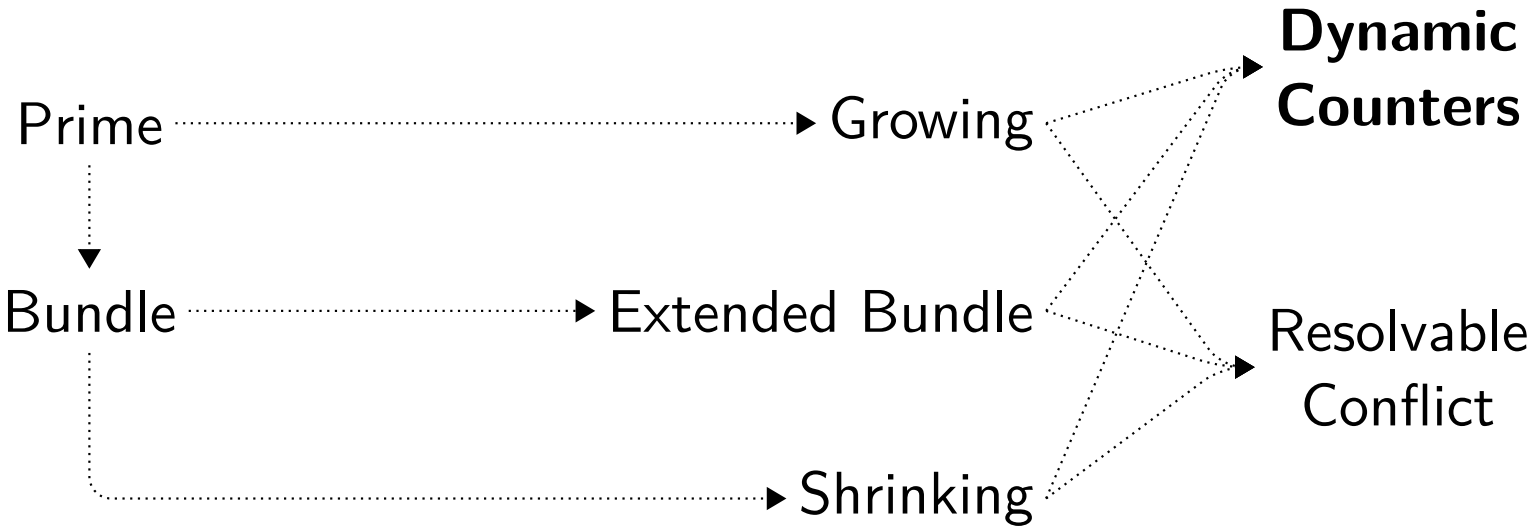
Combining paths:

- extended bundle event structures: disjunctive causality and asymmetric conflict
- dynamic causality event structures: both growing and shrinking causality
- event structures for resolvable conflict: raises enabling to configurations; $\{a_1, \ldots, a_n\} \vdash \{b_1, \ldots, b_m\}$
- higher-order dynamic causality event structures: higher-order dynamic causality relation, and generalisation to sets of causes

Comparison


Prime
Growing
Dynamic
Causality
Bundle
Extended Bundle
Resolvable
Conflict
Shrinking

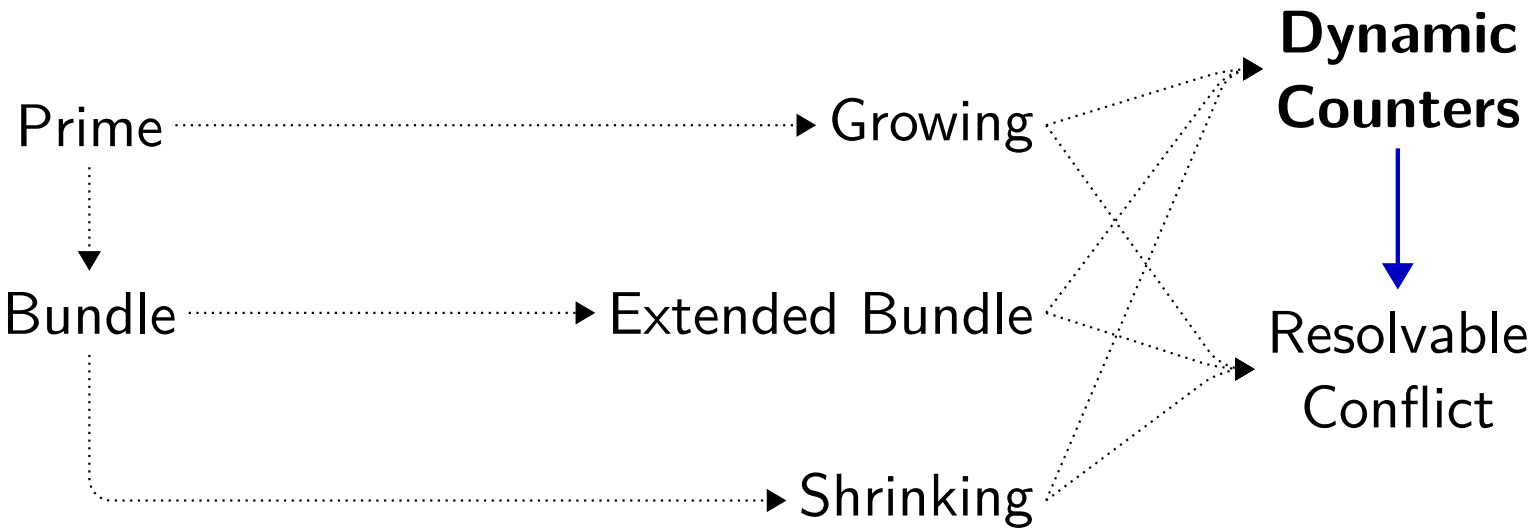
Comparison



Dynamic causality with counters: replaced dynamic causality event structures with a new variant with nice property; the order of events is irrelevant for the resulting causal state

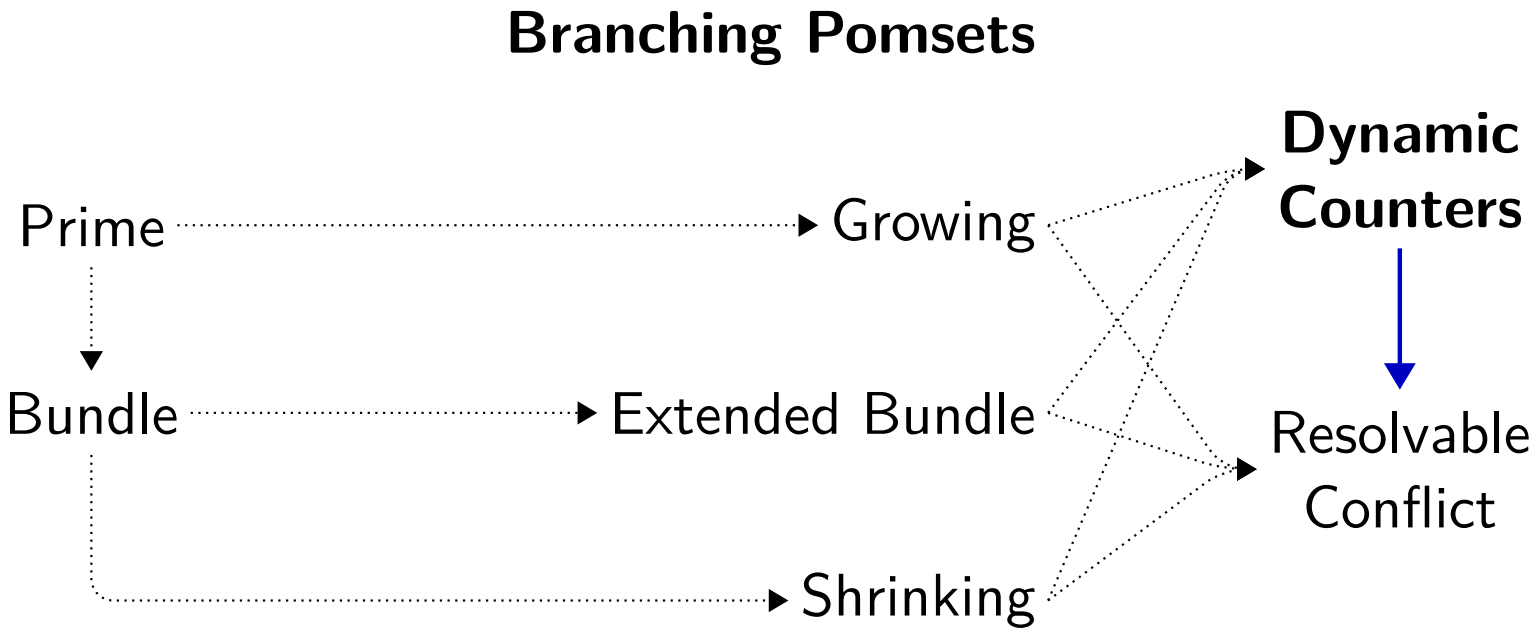
As a result: uniformly defined semantics for all shown classes

Comparison



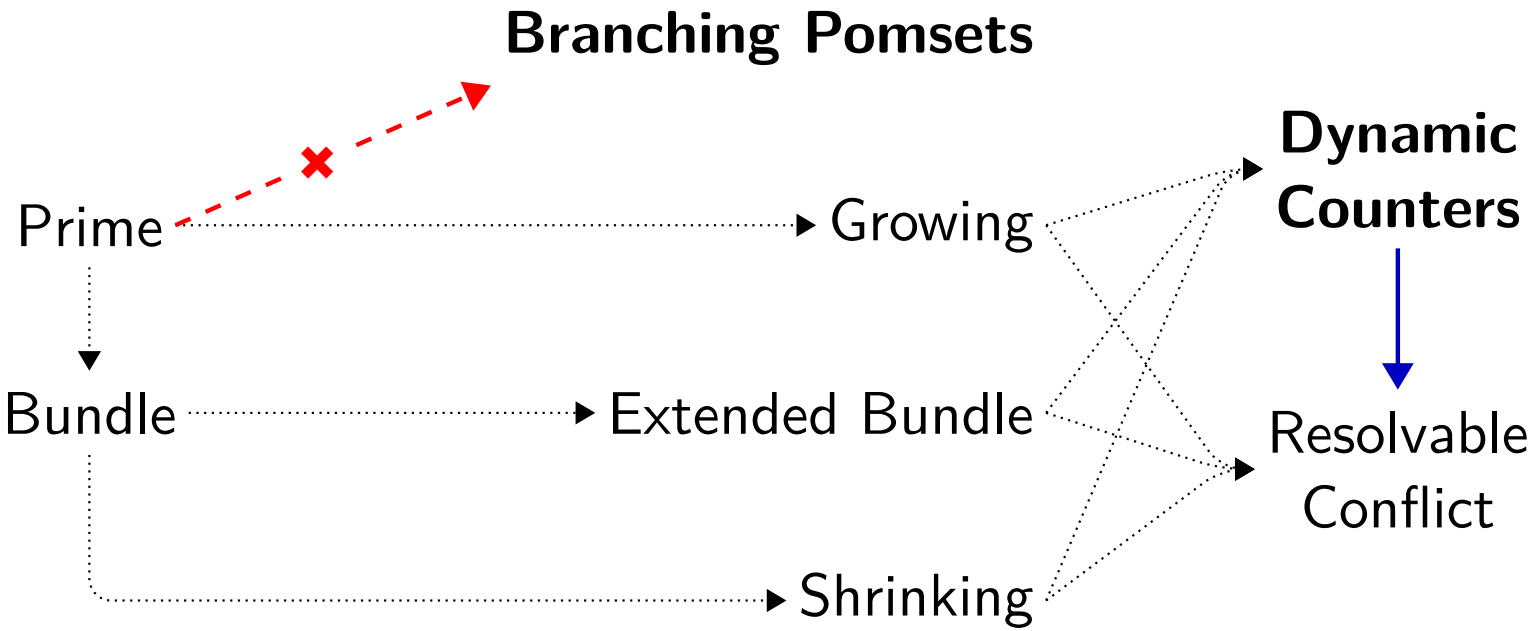
Generic proof: inclusion in event structures for resolvable conflict of any class of event structures where the causal state is order-independent, including dynamic counters

Comparison

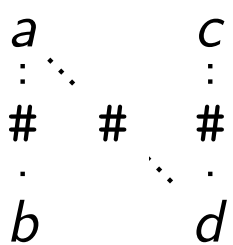


Next up: branching pomsets

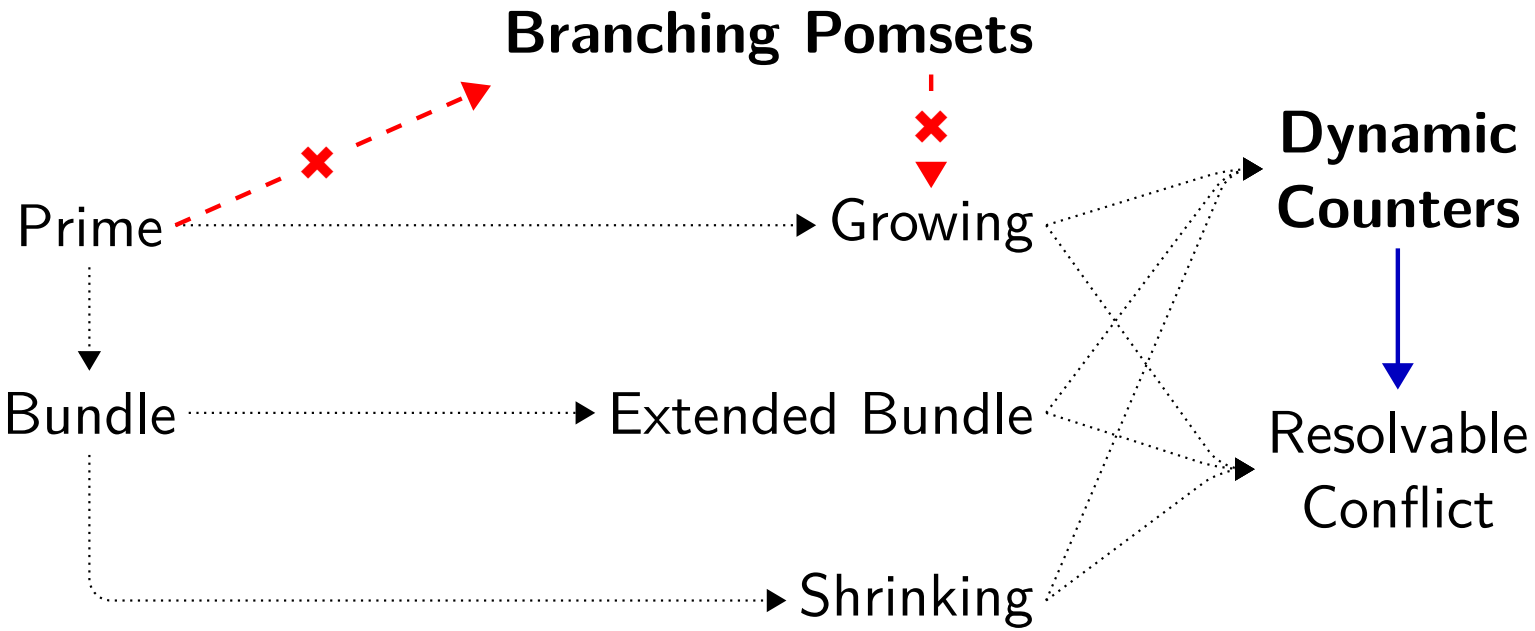
Comparison



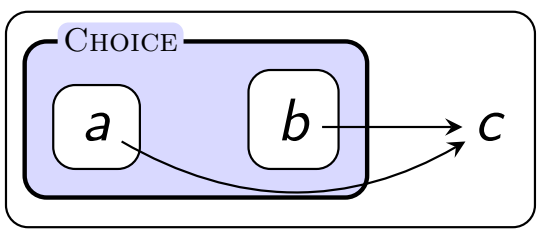
Non-inclusion: not all prime event structures expressible as branching pomsets — would need overlapping boxes



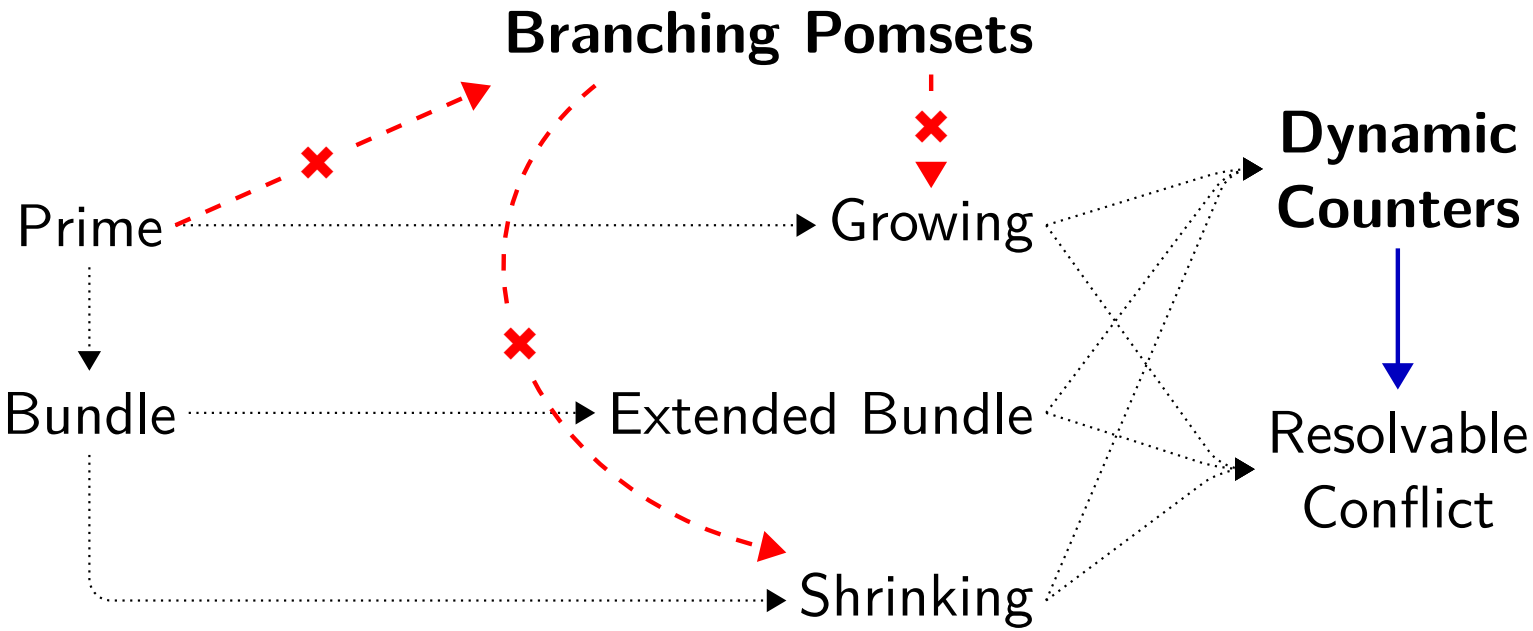
Comparison



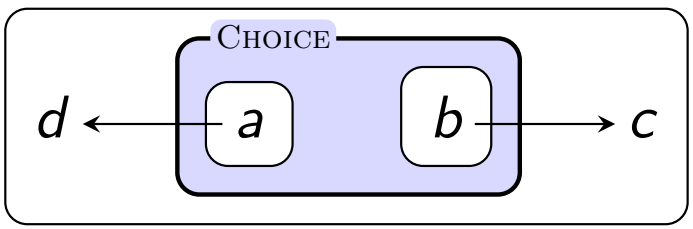
Non-inclusion: not all branching pomsets expressible as growing causality event structures — would need disjunctive causality



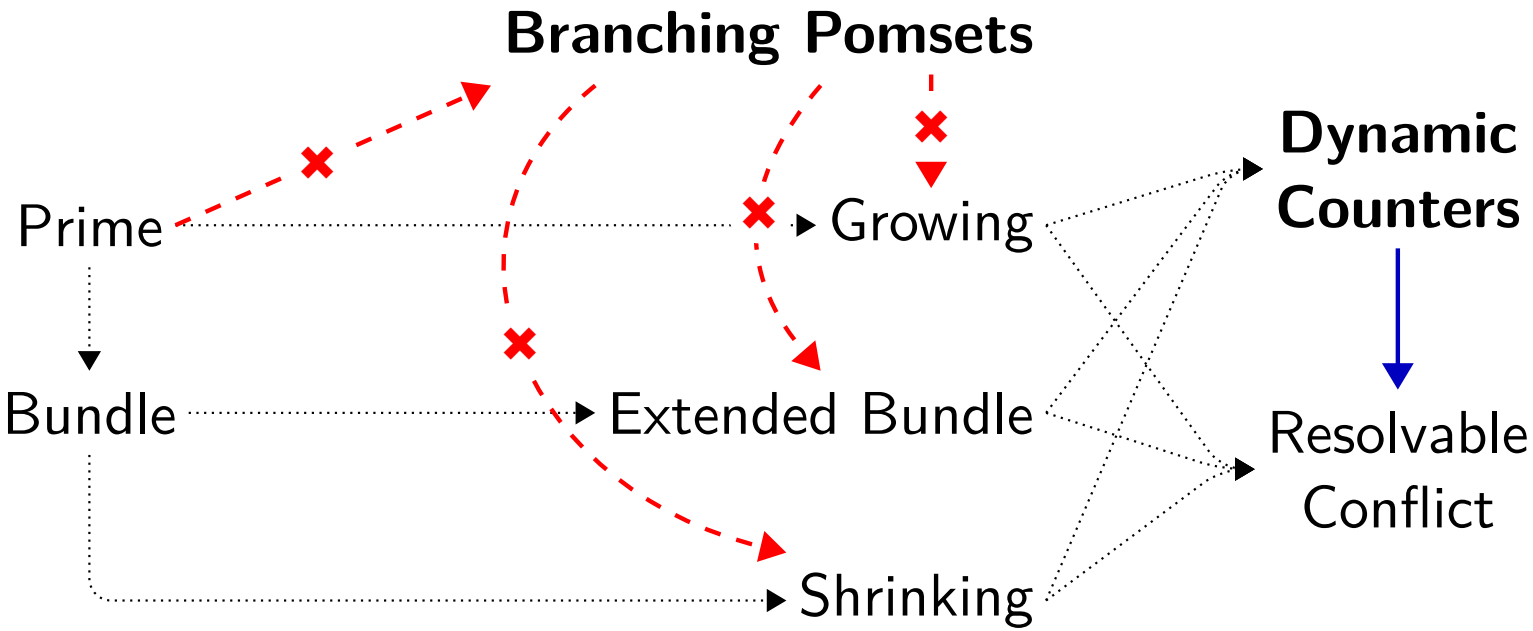
Comparison



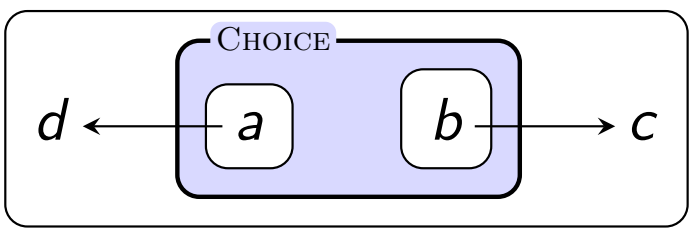
Non-inclusion: not all branching pomsets expressible as shrinking causality event structures — c can be disabled



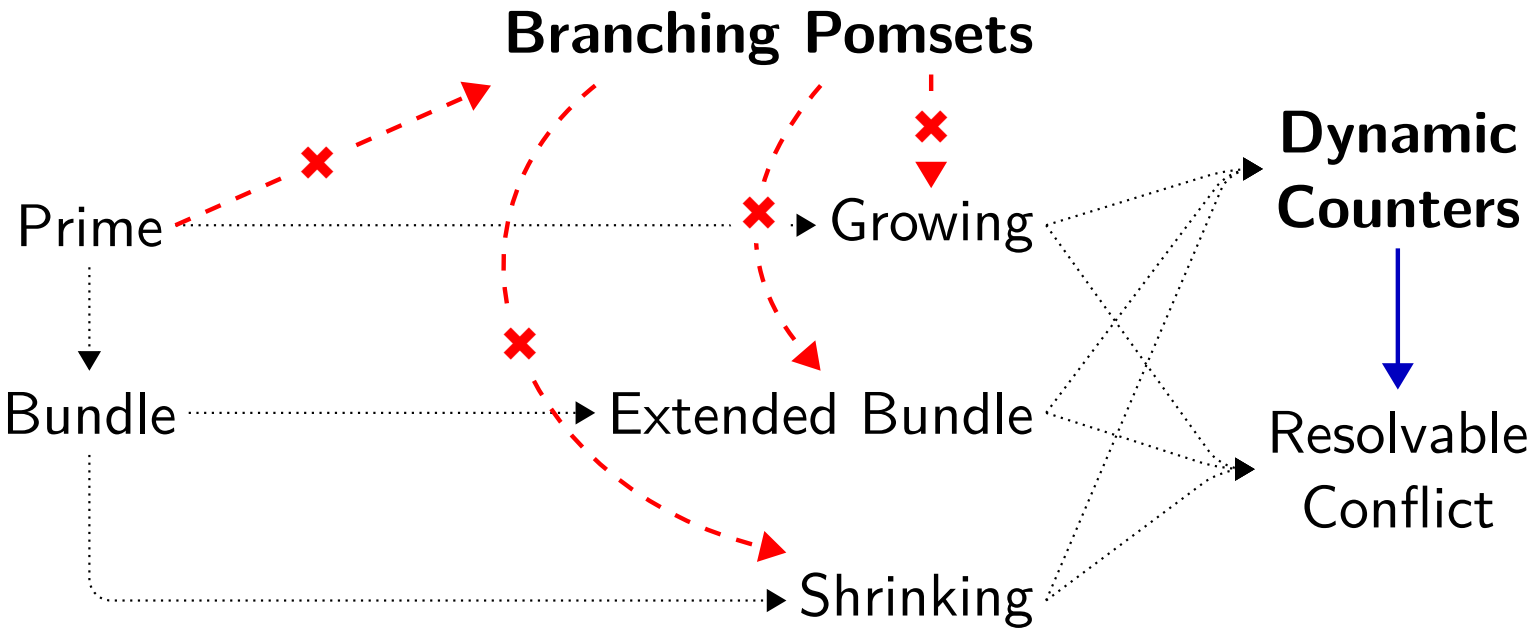
Comparison



Non-inclusion: not all branching pomsets expressible as extended bundle event structures — c can be disabled and then re-enabled

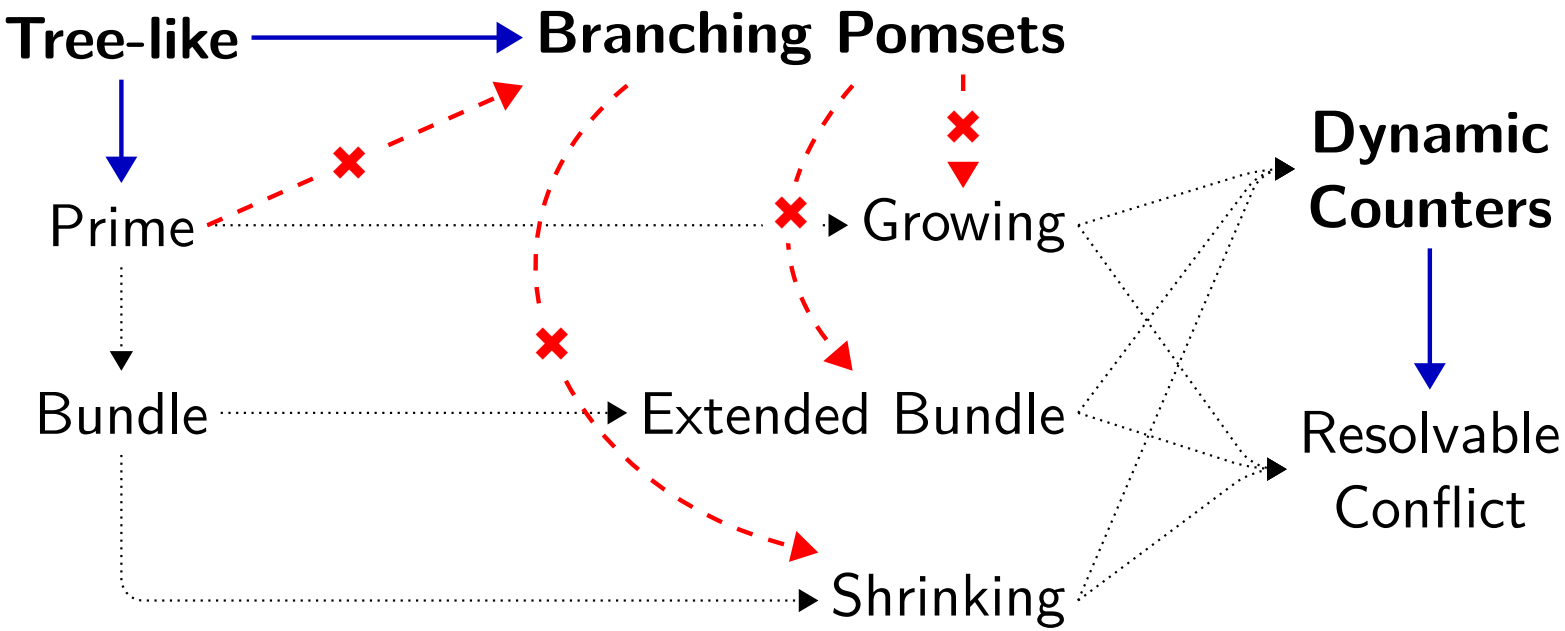


Comparison



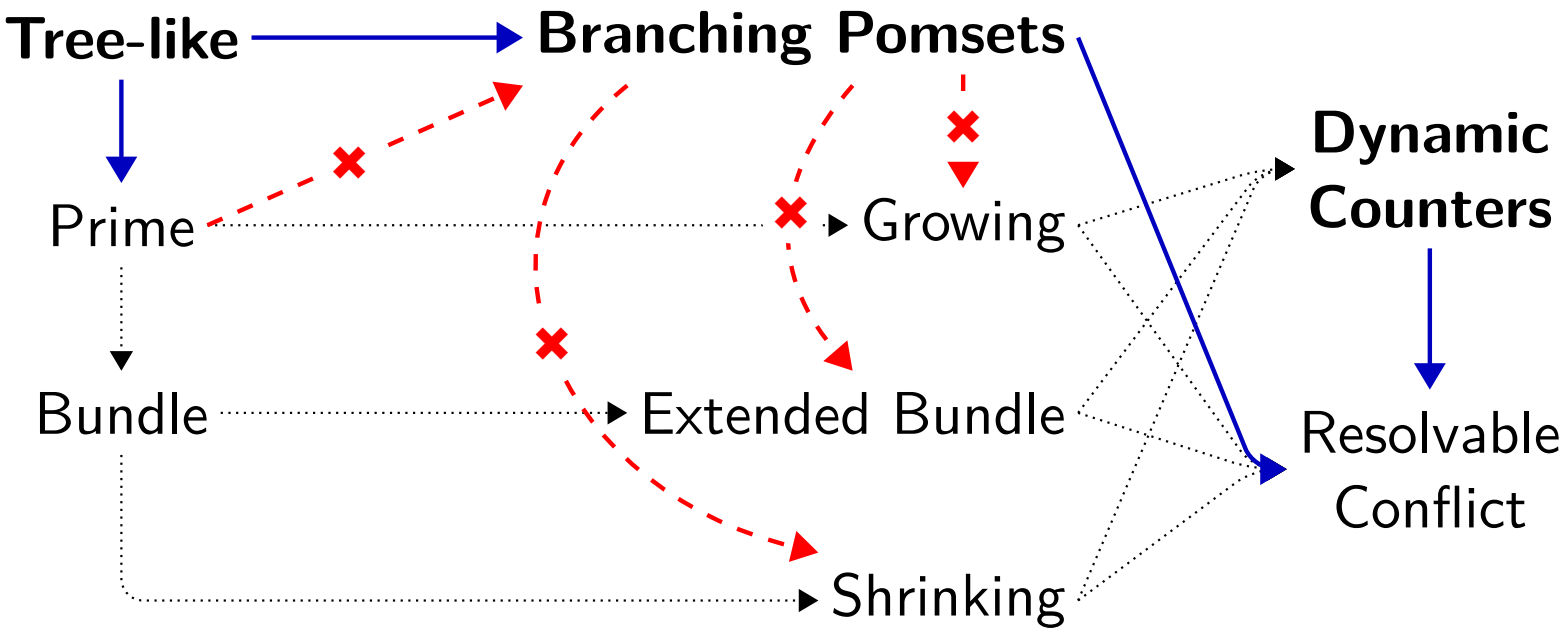
Consequently: branching pomsets incomparable with prime, growing and shrinking causality, and extended bundle event structures (and everything in between)

Comparison



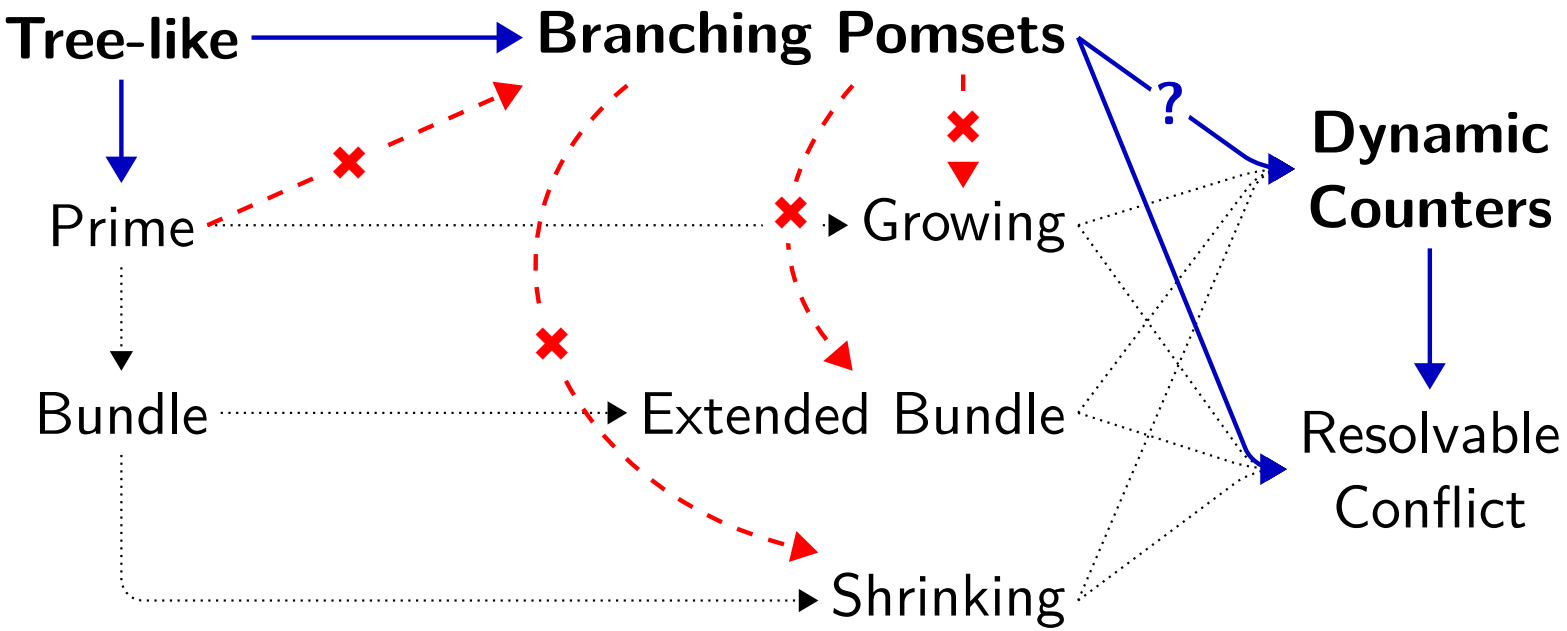
Inclusion: subset of branching pomsets, dubbed *tree-like*, can be expressed as prime event structures

Comparison



Inclusion: same generic proof as for event structures also holds for branching pomsets; they can all be expressed as event structures for resolvable conflict (and consequently as HDESs)

Comparison



Inclusion conjecture: dynamic causality event structures with counters may be powerful enough to express all branching pomsets; no proof yet

Conclusions and future work

Summary

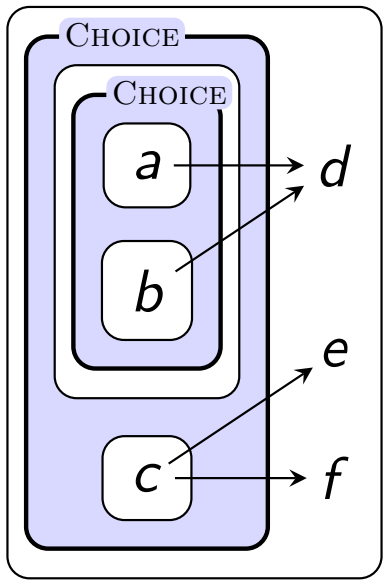
- branching pomsets as a generic model for concurrency
- comparison with various classes of event structures
- interesting behaviour: incomparable with most, included in some more expressive classes of dynamic event structures

Future work

- proving or disproving the dynamic counters conjecture
- study the expressiveness of branching pomsets with overlapping boxes
- expand static analysis of branching pomsets

Dynamic causality event structures

Branching pomsets are **not** included in dynamic causality event structures (without counters).



Higher-dimensional automata

- Event structures for resolvable conflict are equally expressive as Petri nets (van Glabbeek and Plotkin, 2004)
- Petri nets are expressible as HDAs (van Glabbeek, 2006)