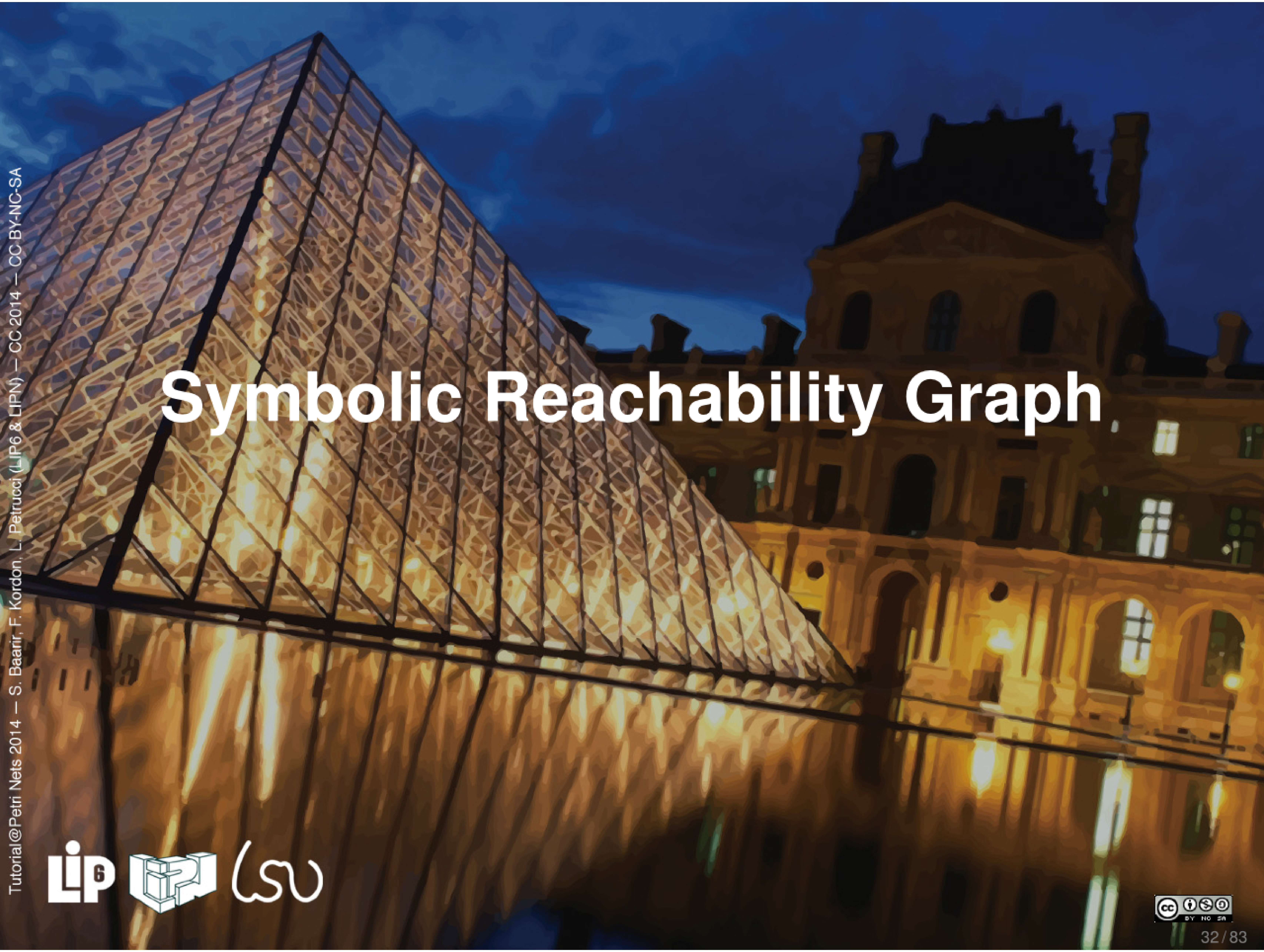


Symbolic Reachability Graph

Tutorial@Petri Nets 2014 — S. Baarir, F. Kordon, L. Petrucci (LIP6 & LIPN) — CC 2014 — CC BY-NC-SA

Introduction

Tutorial@Petri Nets 2014 — S. Baarir, F. Kordon, L. Petrucci (LIP6 & LIPN) — CC 2014 — CC BY-NC-SA

Now, we know:

- how to represent, in symbolic and unique way, the marking classes,
- how to fire from a symbolic marking, a symbolic instance, to obtain the symbolic successor.

Tutorial@Petri Nets 2014 — S. Baarir, F. Kordon, L. Petrucci (LIP6 & LIPN) — CC 2014 — CC BY-NC-SA

Now, we know:

- how to represent, in symbolic and unique way, the marking classes,
- how to fire from a symbolic marking, a symbolic instance, to obtain the symbolic successor.

We are ready to derive an algorithm to construct the symbolic reachability graph.

SRG construction Algorithm

SRG_Construction$(N = \langle P, T, C, W^-, W^+, M_0 \rangle)$

$SRG.Q = \{\hat{M}_0\}$; $SRG.\delta = \emptyset$;

$SRG.q_0 = \hat{M}_0$; $sStates = \{\hat{M}_o\}$:

While (sStates <> $\emptyset$) {

  $\hat{s}$ = pick a sstate in sStates ;

  sStates = sStates \ $\{\hat{s}\}$;

  for each $t \in T, \hat{c} \in \hat{C}(t)$ {

    if $(\hat{s}[(t, \hat{c})\rangle)$ {

      $\hat{s}[(t, \hat{c})\rangle \hat{ns}$;

      if $(\hat{ns} \notin SRG.Q)$ {

        $SRG.Q = SRG.Q \cup \{\hat{ns}\}$;

        $sStates = sStates \cup \{\hat{ns}\}$;

      }

      $SRG.\delta = SRG.\delta \cup \{(\hat{s}, \hat{ns})\}$;

      $SRG.\lambda(\hat{s}, \hat{ns}) = (t, \hat{c})$;

    }

  }

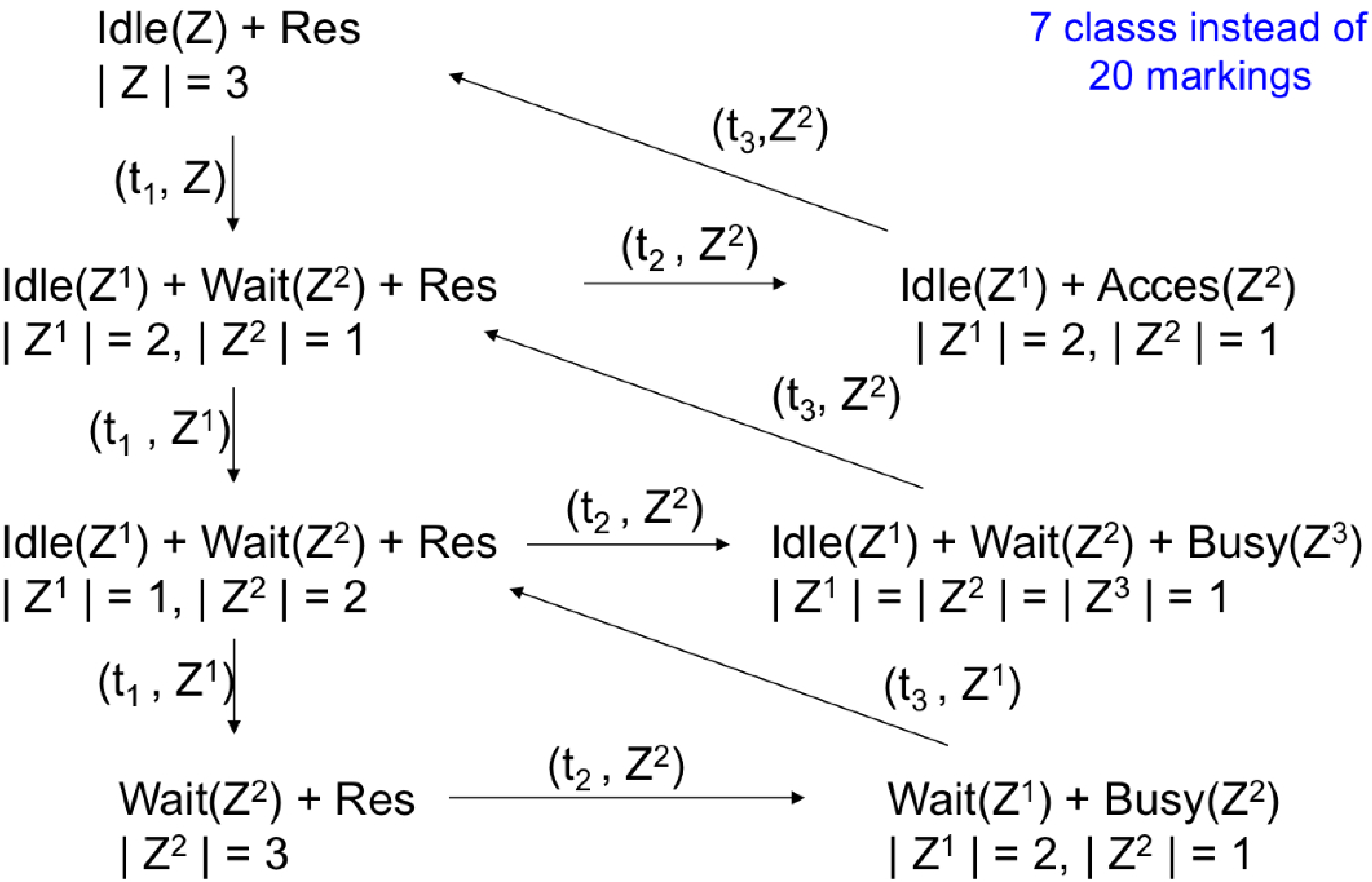
}

return SRG;

Tutorial@Petri Nets 2014 — S. Baarir, F. Kordon, L. Petrucci (LIP6 & LIPN) — CC 2014 — CC BY-NC-SA

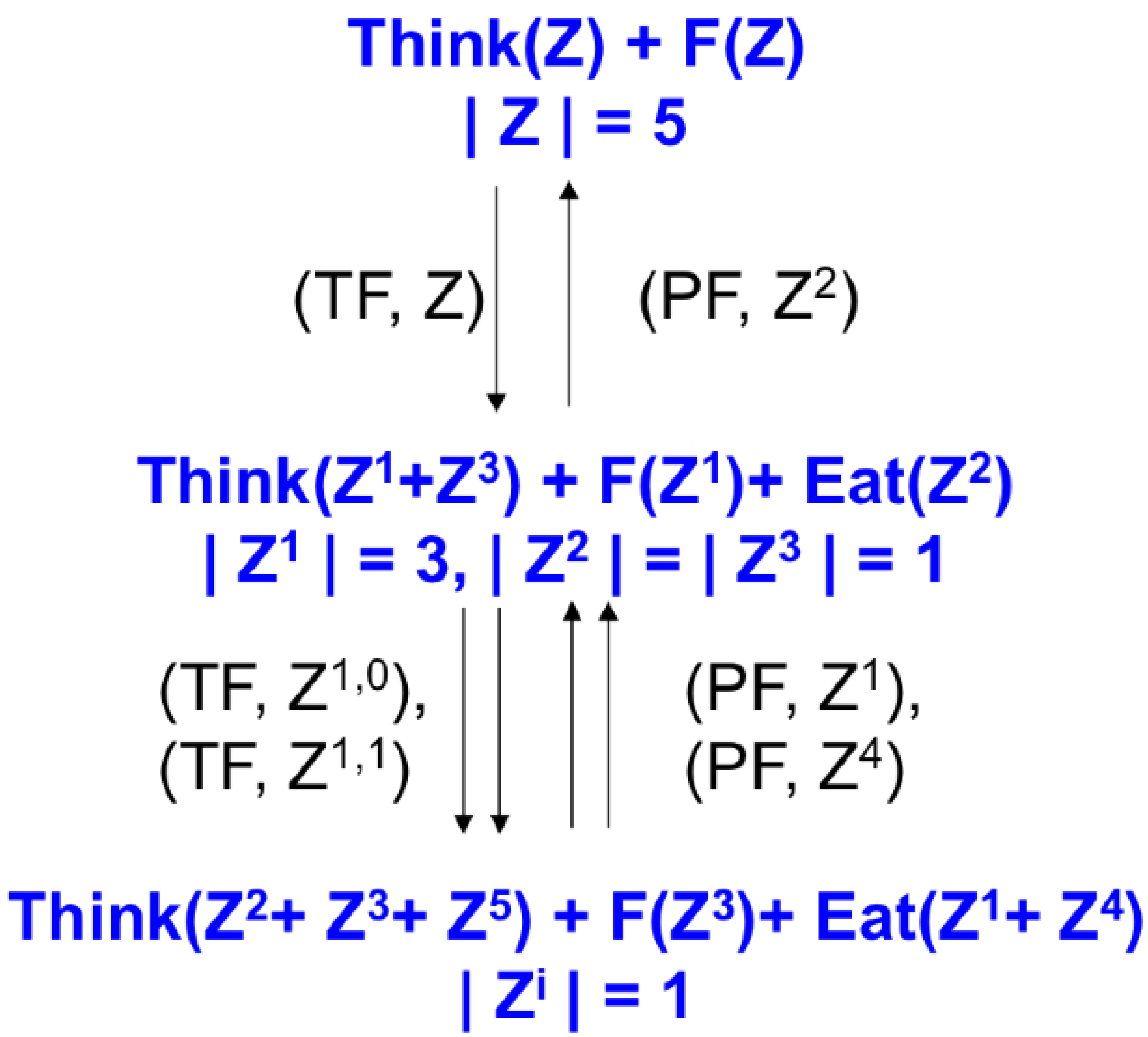
Example: SRG of the critical section access model

7 classs instead of 20 markings



Tutorial@Petri Nets 2014 — S. Baarir, F. Kordon, L. Petrucci (LIP6 & LIPN) — CC 2014 — CC BY-NC-SA

Example: SRG of the dining philosophers problem



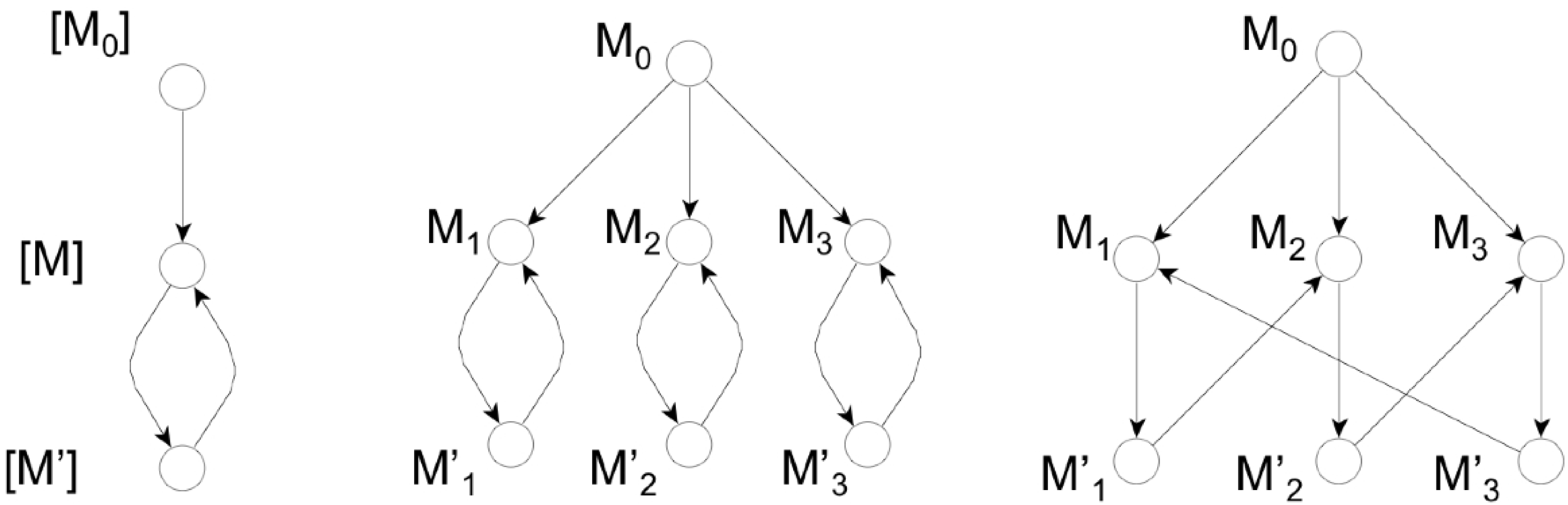
3 symbolic markings instead of 11 markings

What does the Symbolic Reachability Graph preserve?

- Each marking represented by a class (a symbolic marking) is reachable.
- Each reachable marking is represented by a class.
- Each firing sequence of the RG is represented in the SRG.
- To each sequence of the symbolic graph corresponds a sequence of the RG.

Tutorial@Petri Nets 2014 — S. Baarir, F. Kordon, L. Petrucci (LIP6 & LIPN) — CC 2014 — CC BY-NC-SA

- We cannot distinguish the following situations:



Tutorial@Petri Nets 2014 — S. Baarir, F. Kordon, L. Petrucci (LIP6 & LIPN) — CC 2014 — CC BY-NC-SA

Conclusion

- So far, the approach presented imposes that all objects of the same class behave identically.
  - A class groups a set of objects that have the same nature.
  - The obtained reduction, SRG vs. RG, is maximal.

- How to deal with the case where objects have the same nature, *but with potentially different behaviours*?
  - Example: a class that represents a set of processors divided in two subsets: fast and slow.

Tutorial@Petri Nets 2014 — S. Baarir, F. Kordon, L. Petrucci (LIP6 & LIPN) — CC 2014 — CC BY-NC-SA

Conclusion

- So far, the approach presented imposes that all objects of the same class behave identically.
  - A class groups a set of objects that have the same nature.
  - The obtained reduction, SRG vs. RG, is maximal.
- How to deal with the case where objects have the same nature, *but with potentially different behaviours*?
  - Example: a class that represents a set of processors divided in two subsets: fast and slow.
- Use of static subclasses...
  - Each class is partitioned into cells, called static subclasses, where the objects of the same cell behave identically.
  - Symmetries of net extends easily as follows... (next sequence)

Tutorial@Petri Nets 2014 — S. Baarir, F. Kordon, L. Petrucci (LIP6 & LIPN) — CC 2014 — CC BY-NC-SA