A Biologically Inspired Model with Fusion and Clonation of Membranes

Giorgio Delzanno* and Laurent Van Begin**

*DISI, University of Genova

**Universite' Libre de Bruxelles

UC 2008, Wien, August 26, 2008

Motivations and goals

- New computational models with biologically inspired concepts, data structures, and operations
  - membrane, fusion, clonation/division
- Expressive power of the new models
- Verification methods for their qualitative analysis
- Comparison with standard concurrency models like Petri Nets and Process Algebra

Contents of the Talk

- Basics on P-systems
- Extension with fusion and creation
- Computational properties:
  - Reachability, Boundedness, and Coverability
- (Un)decidability results
- Conclusions

P-systems

Abstract model of the living cell

G. Paun. Computing with membranes, JCSS 61(1), 2000

A P-system consists of

Membranes

- Hierarchically structured set of named containers
- Membranes contains multisets of objects

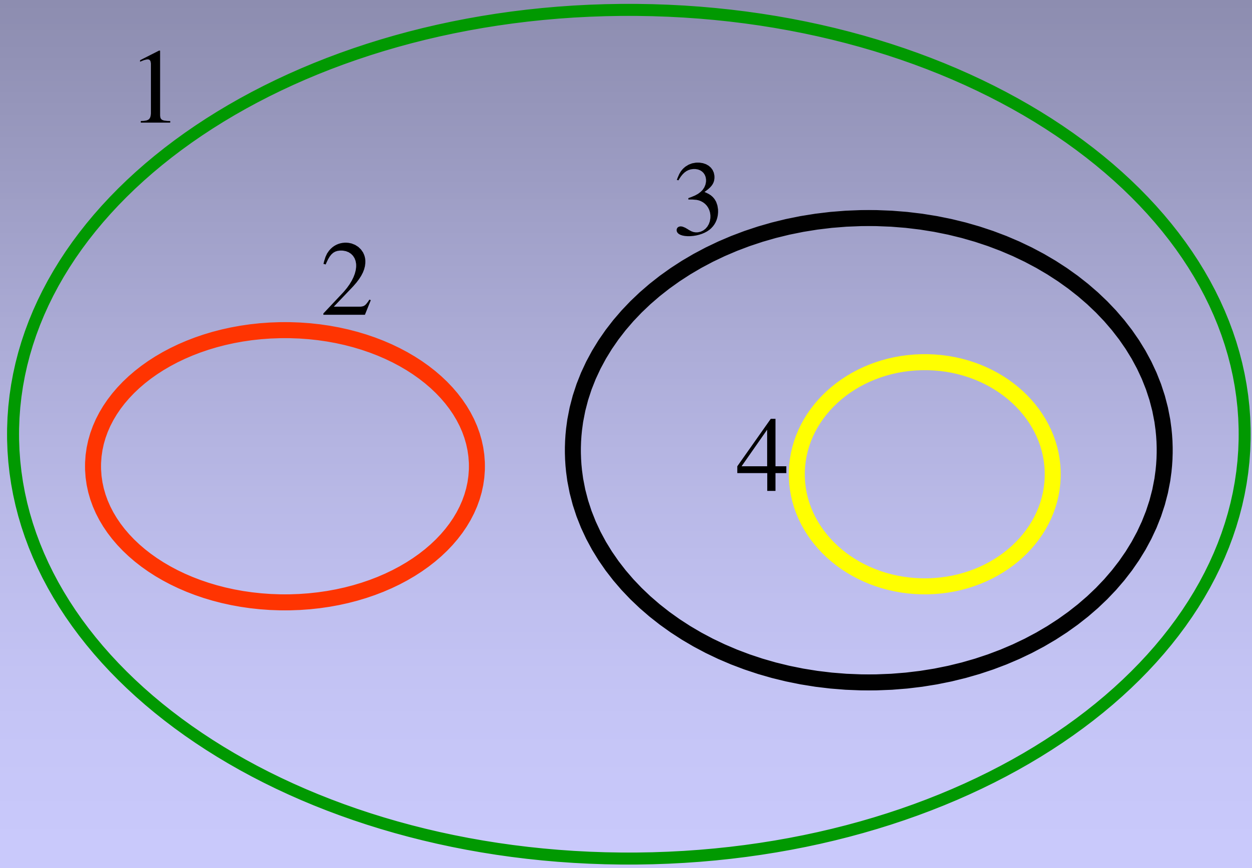
Objects

- Elementary particles (symbol objects)
- Strings (word objects)

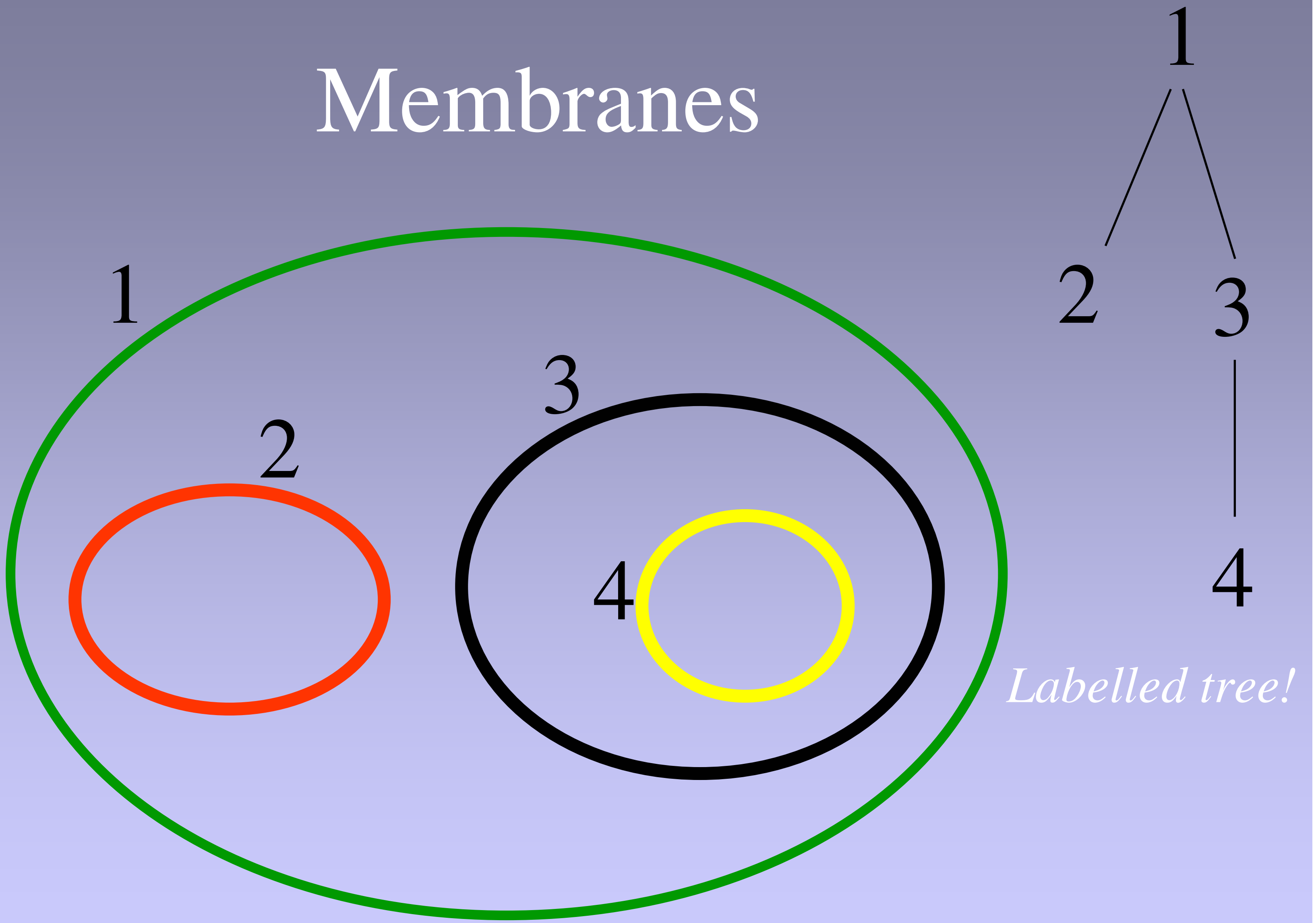
Rules

- To distribute objects across membranes

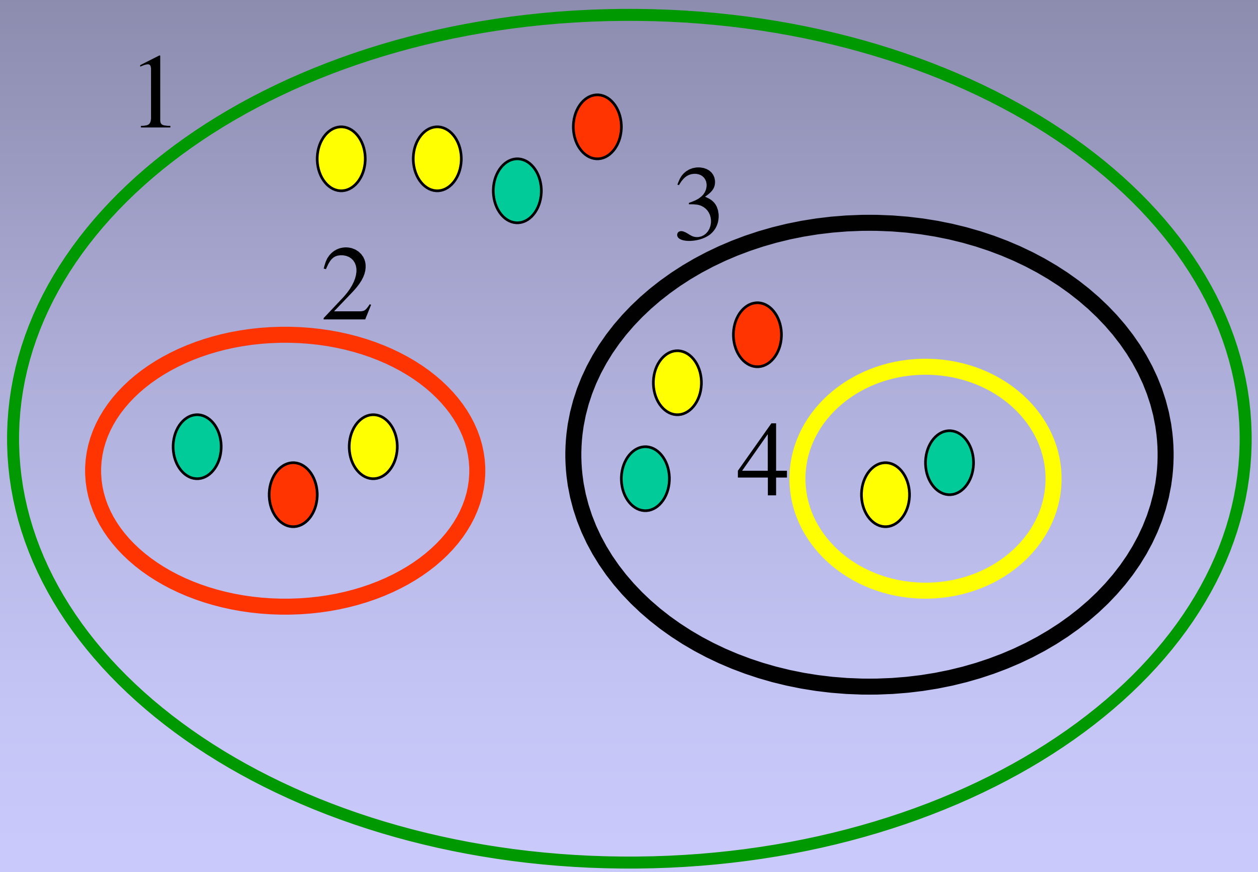
Membranes



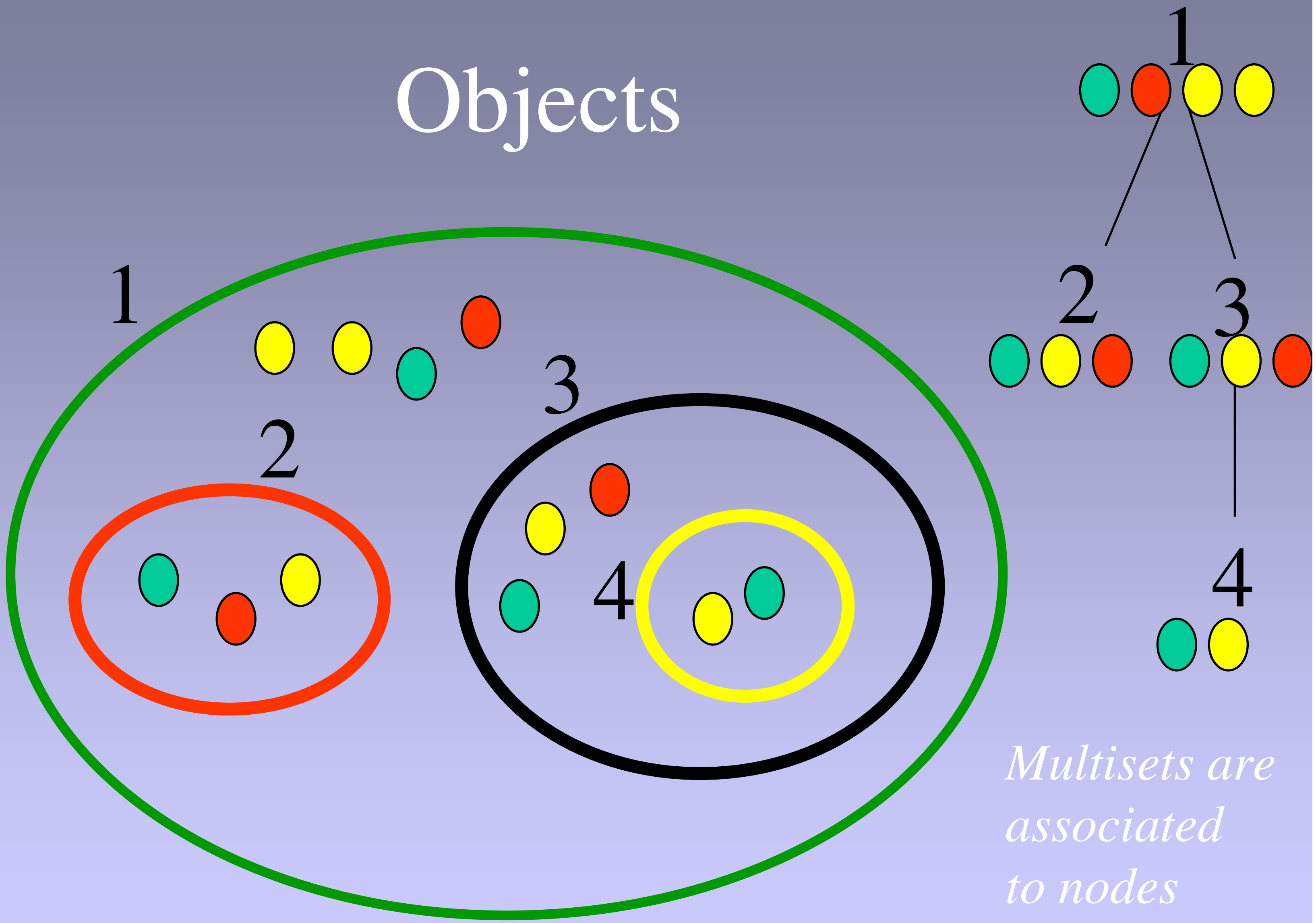
Membranes



Objects



Objects



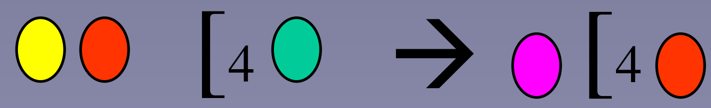
Multisets are associated to nodes

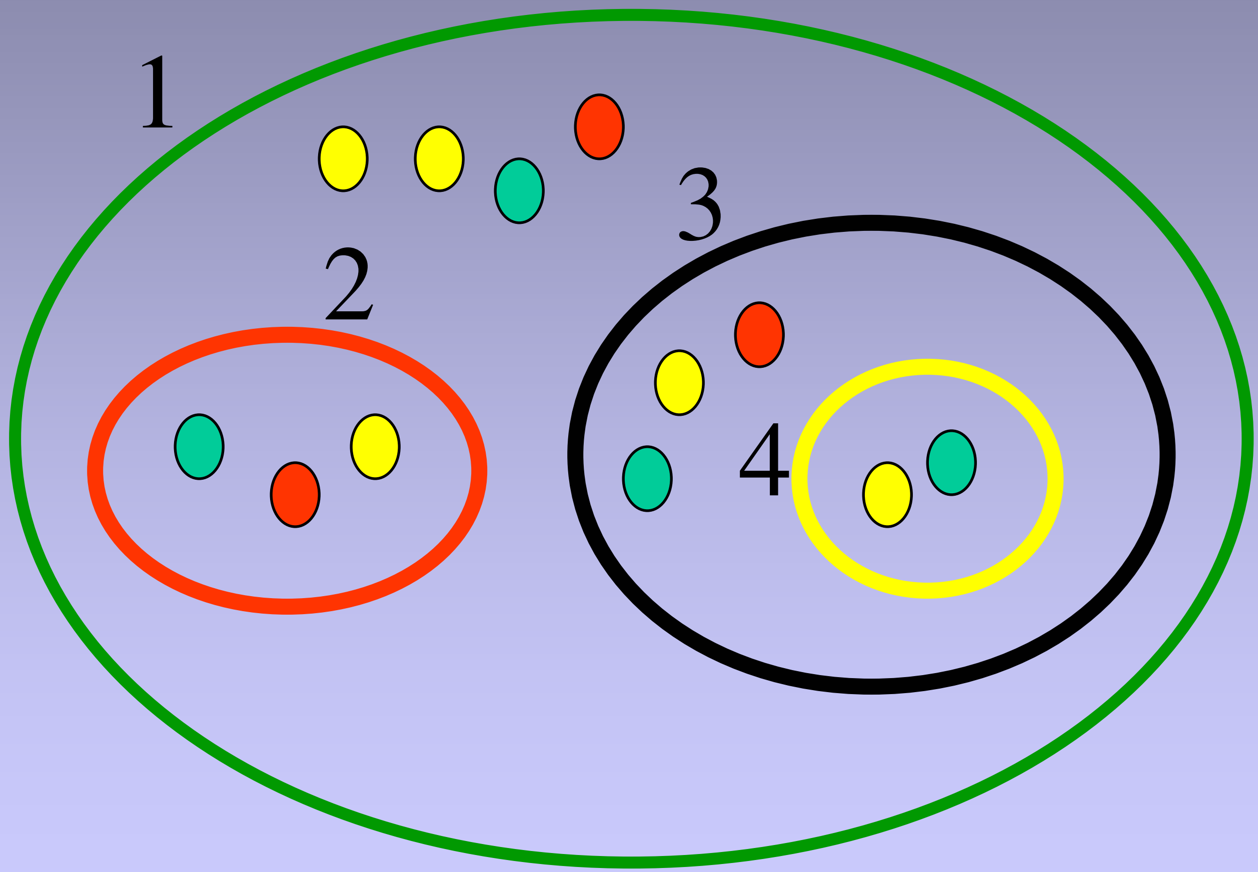
Boundary Rules

$$w \ [_i \ v \rightarrow z \ [_i \ t$$

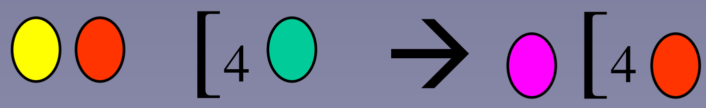
- w (v) = objects that are consumed
  outside (inside) membrane i
- z (t) = objects that are created
  outside (inside) membrane i

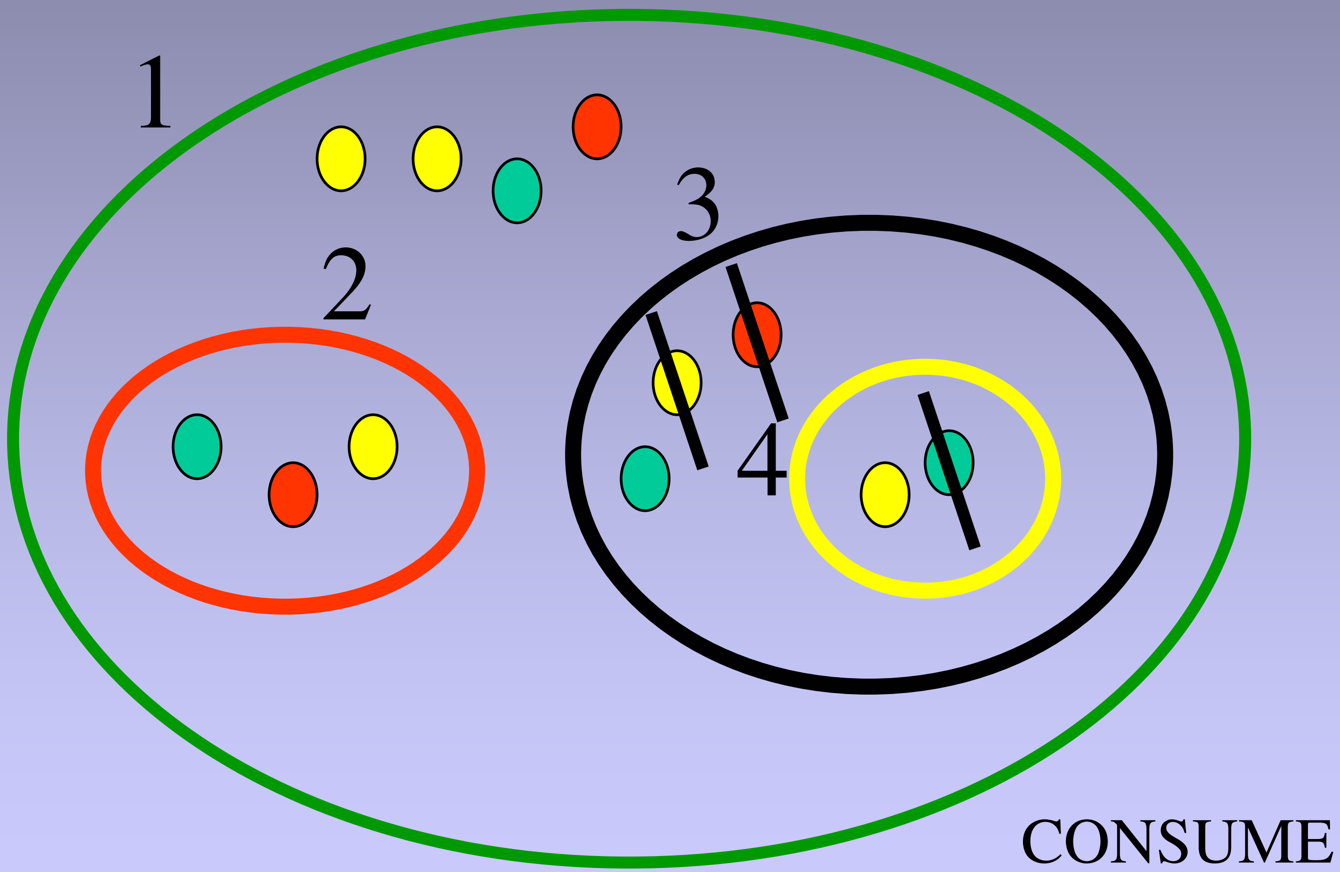
Example



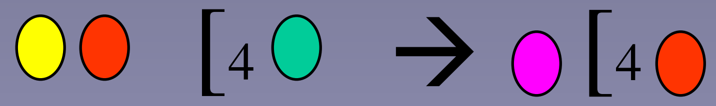


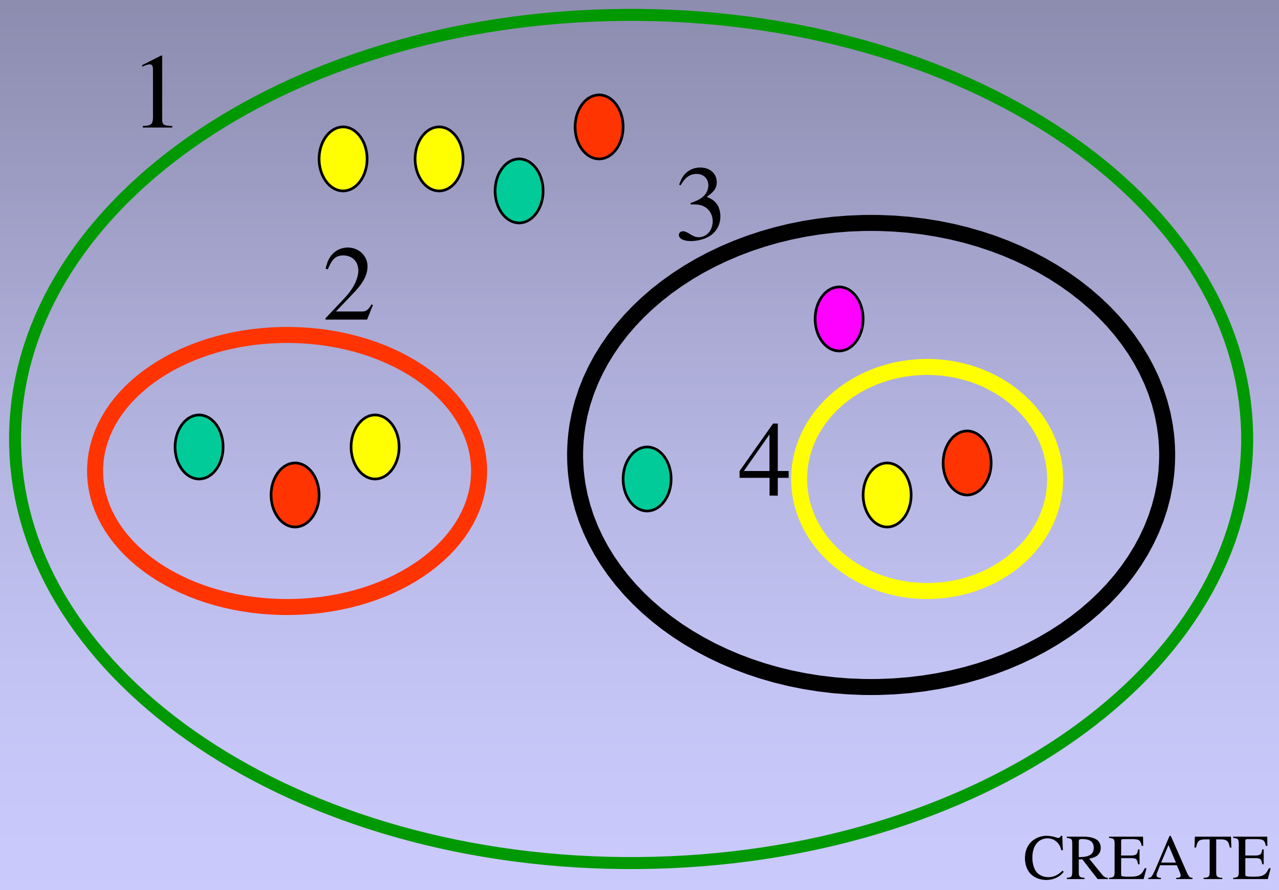
Example





Example





Computational Power

- P-systems with symbol objects and maximal parallelism are Turing powerful

- P-systems with symbol objects and interleaving semantics (PB systems) are equivalent to Petri nets

[BM02] Bernardini-Manca. P systems with boundary rules. WMC 2002.
[DF03] Dal Zilio-Formenti.On the dynamics of PB systems WMC 2003

PBFC systems

PB-systems with Fusion and Creation Rules

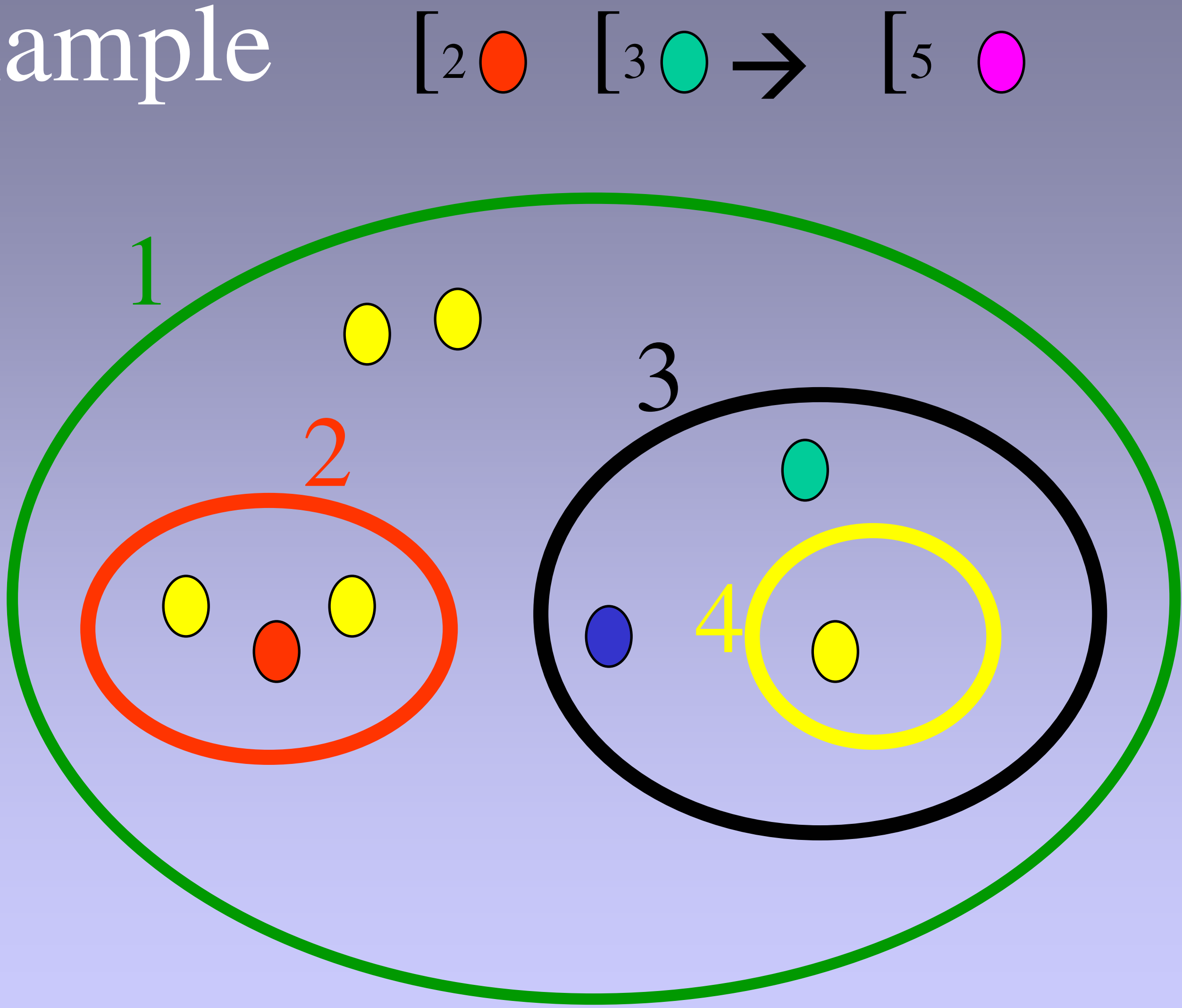
Fusion

Fusion Rules

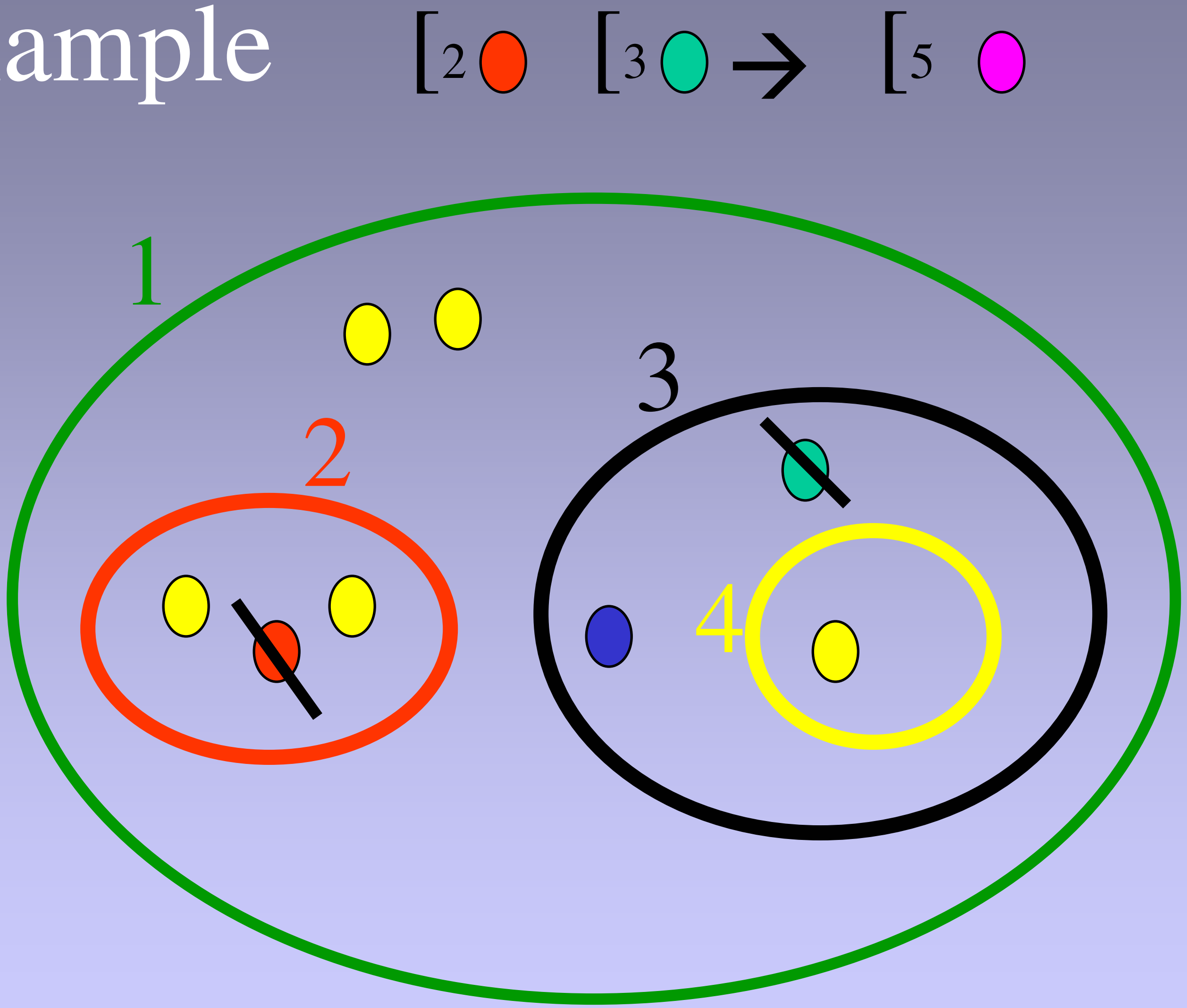
$[_i\ w \quad [_j\ v \rightarrow [_k\ z$

- Membrane i and j are merged into membrane k
- w (v) = objects that are consumed inside membrane i and j
- z = objects that are created inside membrane k

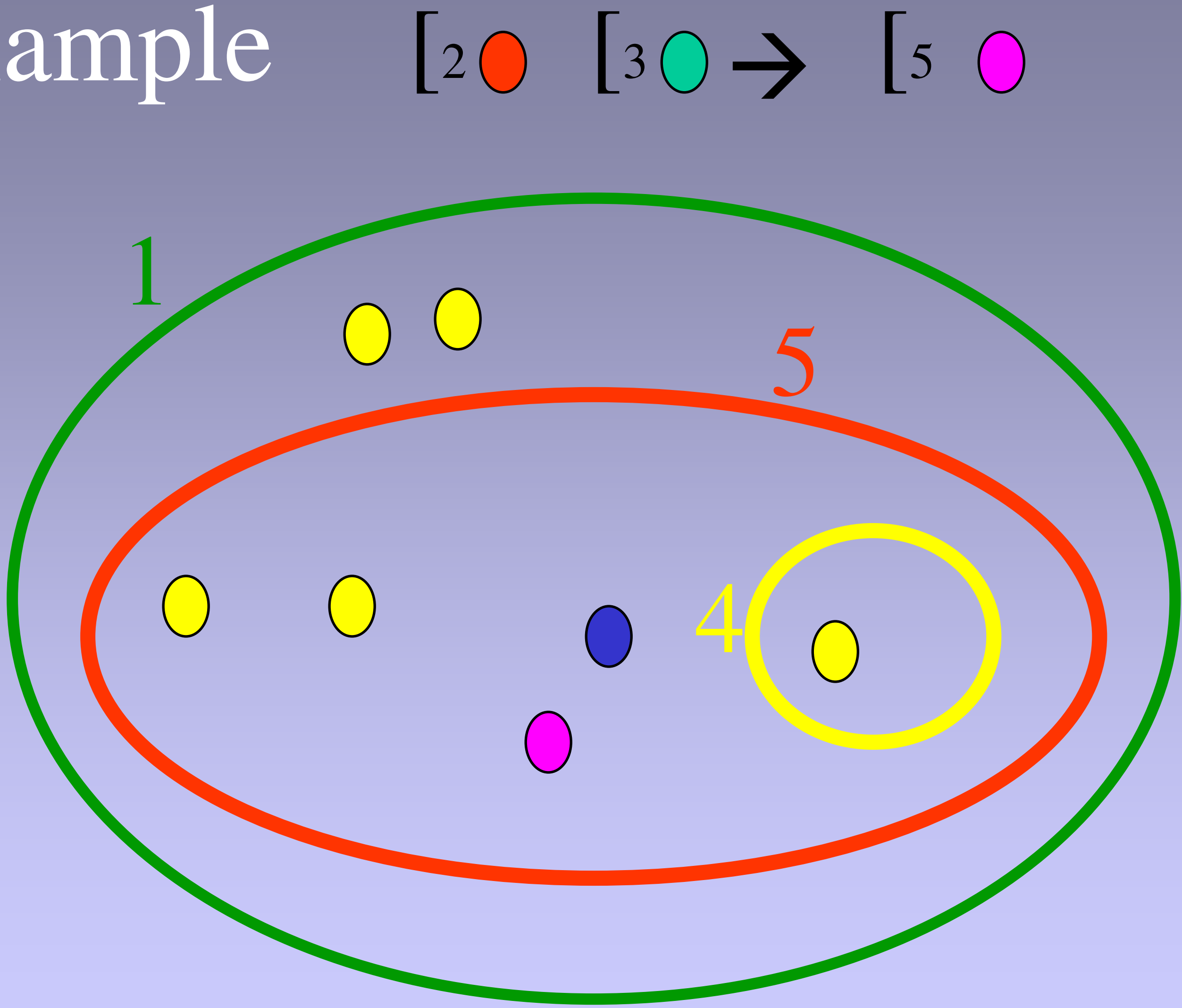
Example



Example



Example



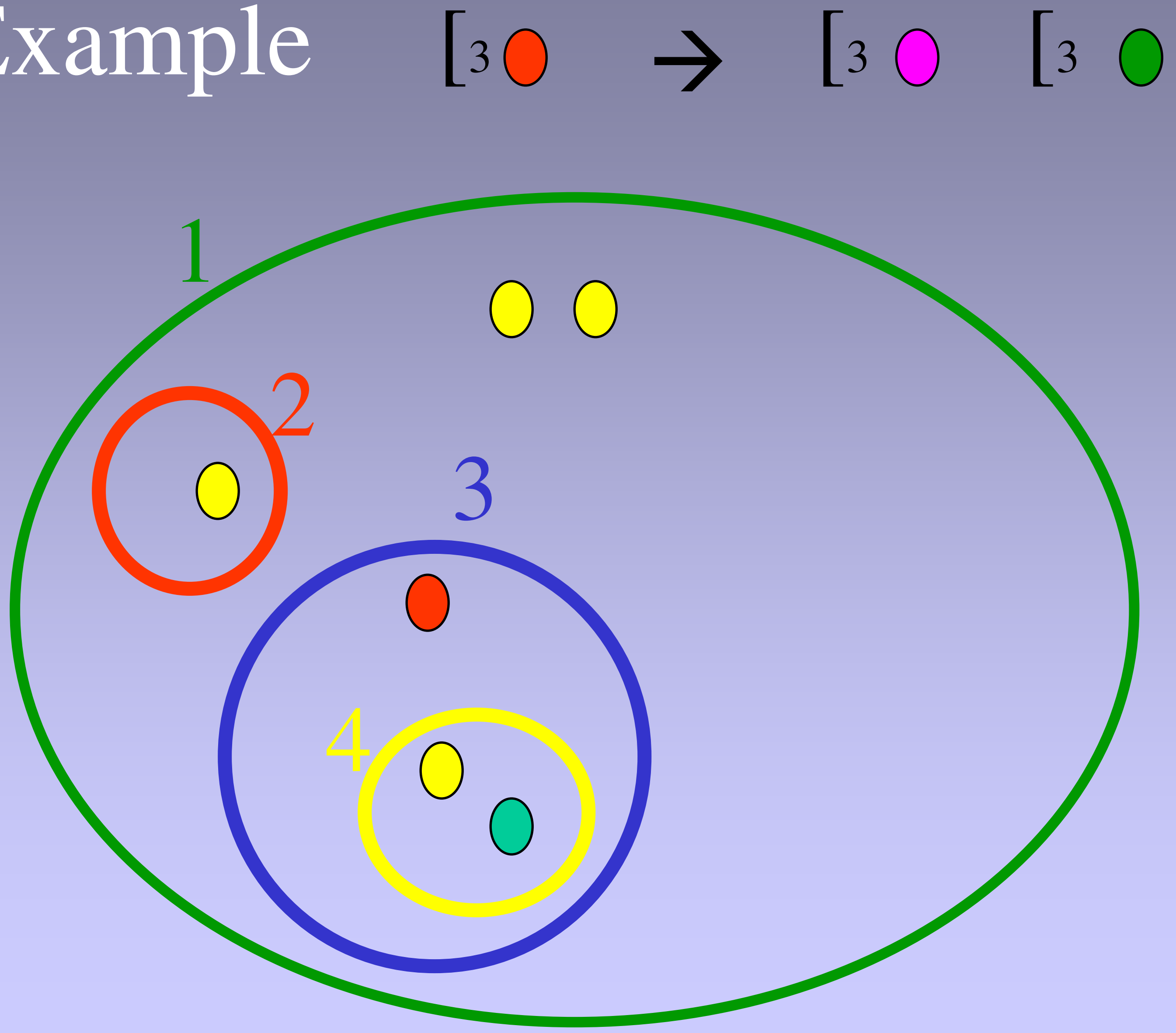
Clonation

Clonation Rules

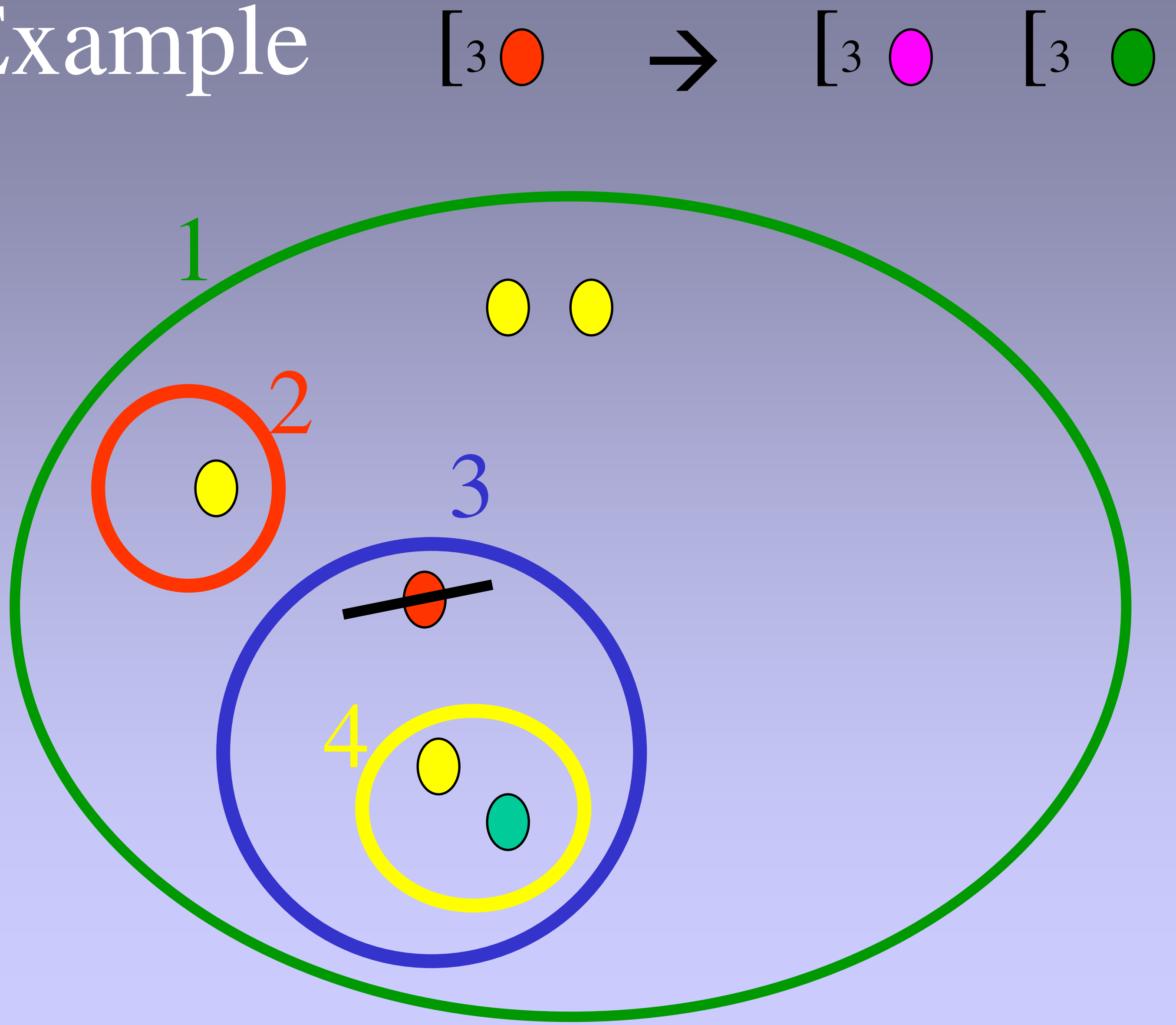
$[_i\ w \rightarrow [_i\ v \quad [_i\ z$

- Membrane i is duplicated
- w = objects that are consumed inside membrane i
- v and z = objects that are created inside the two copies

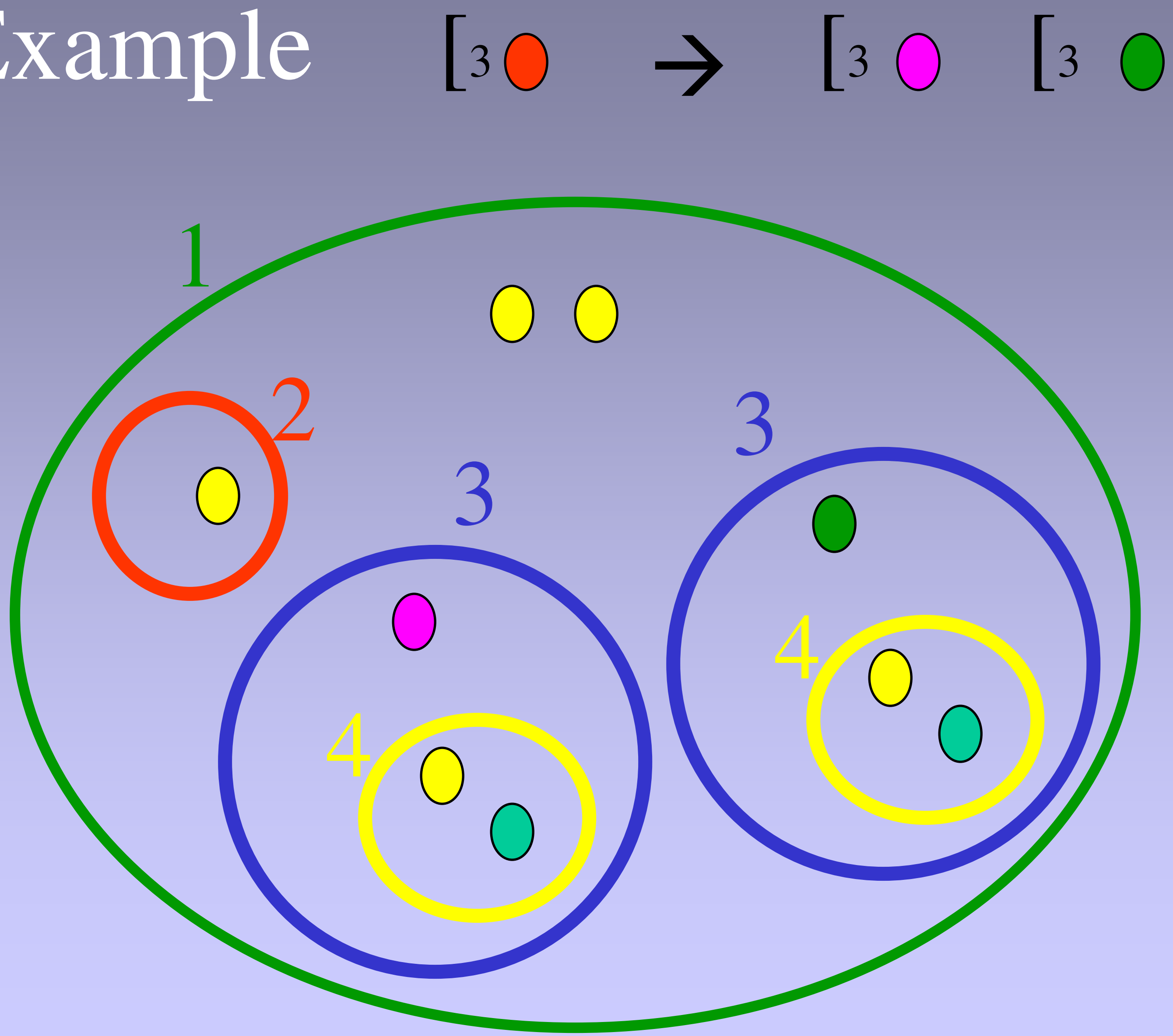
Example



Example



Example



Reachability Problem (RP)

Definition of RP

- Assume a PB system with boundary, fusion and creation rules
- Fix two configurations C0 and C1

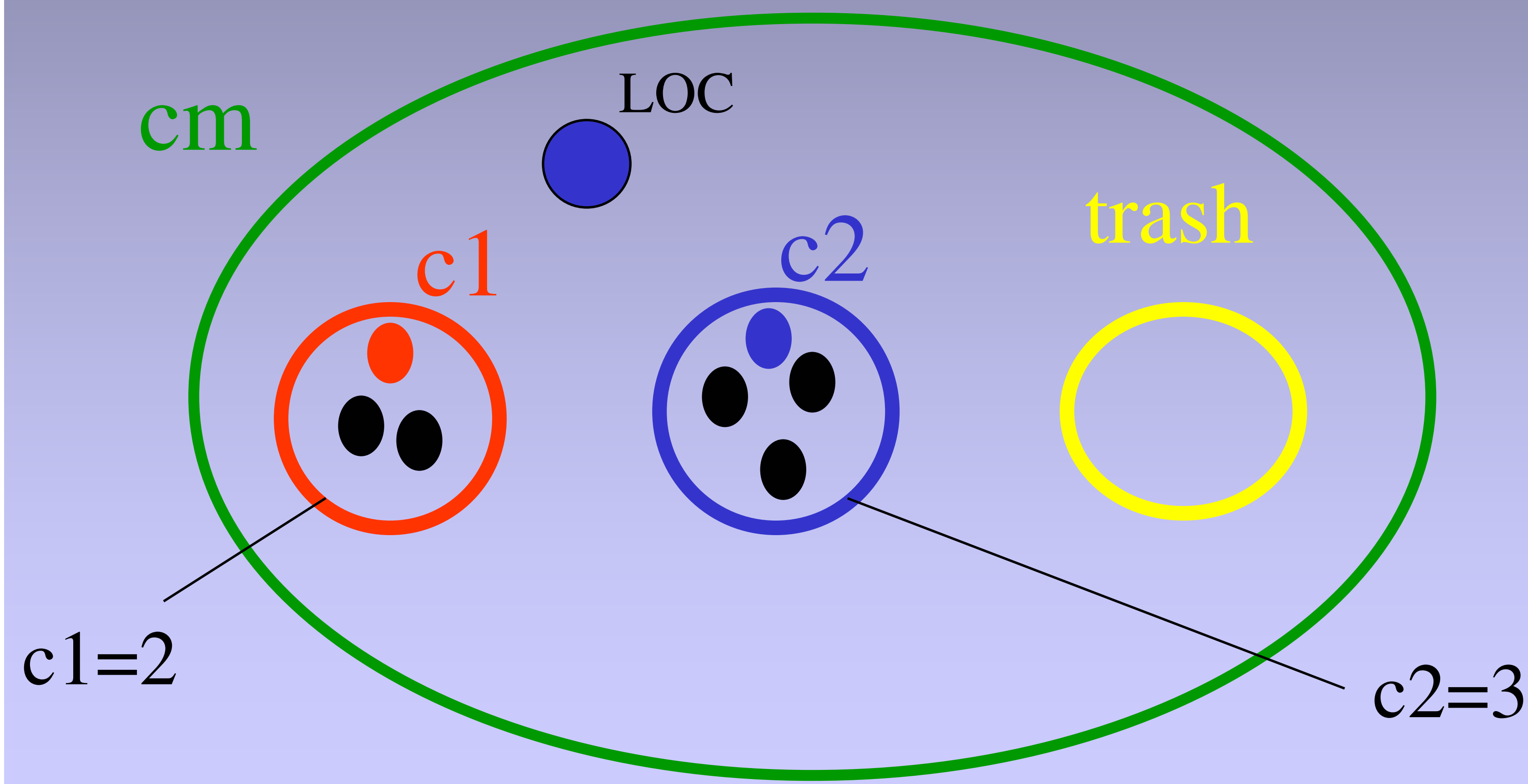
(RP) Is C1 reachable from C0 by firing rules a finite number of times?

Our Result

- RP is **undecidable** in presence of fusion and clonation
- ***Remark***: RP is **decidable** for PB systems (only boundary rules)

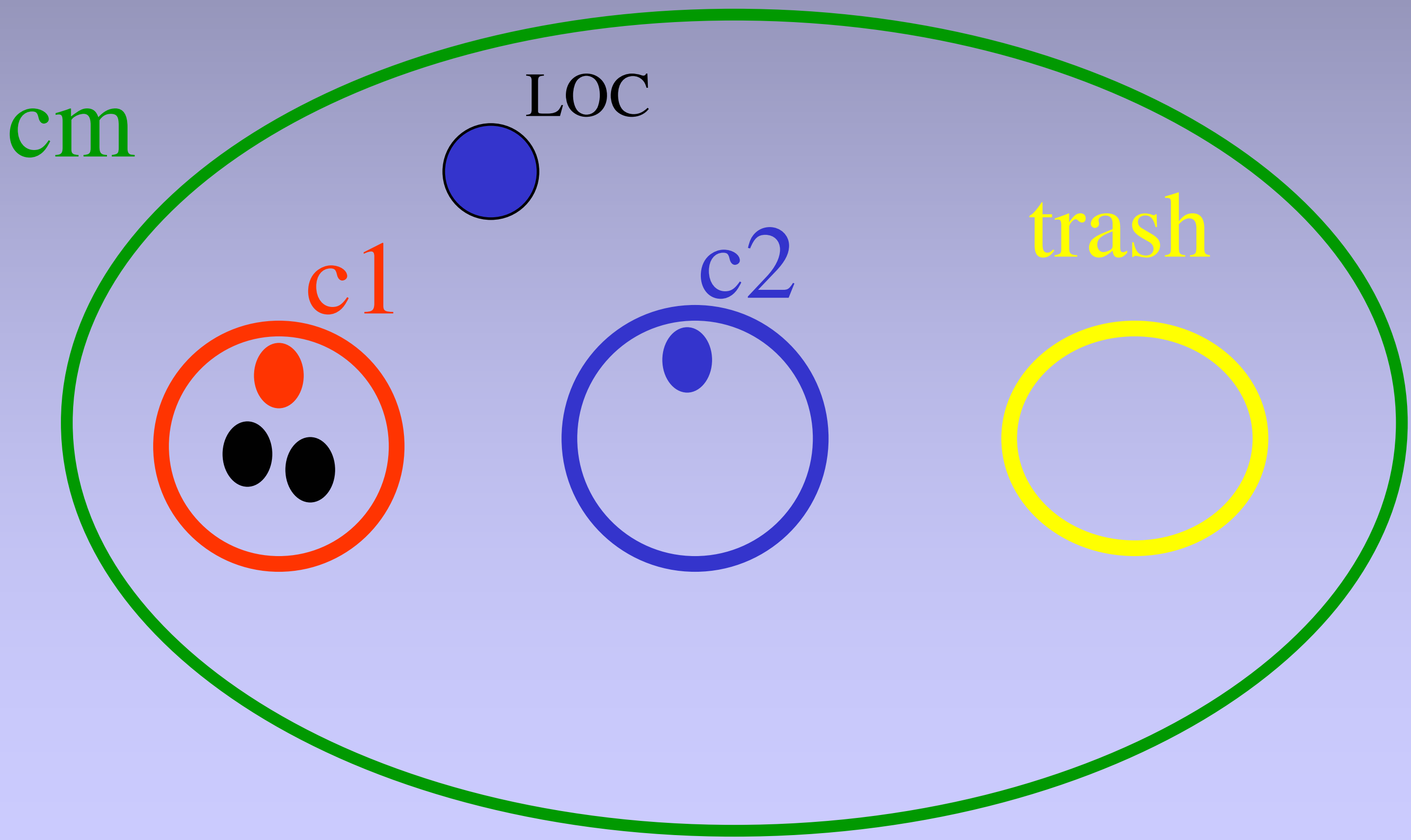
Undecidability proof

- We weakly simulate counter machines



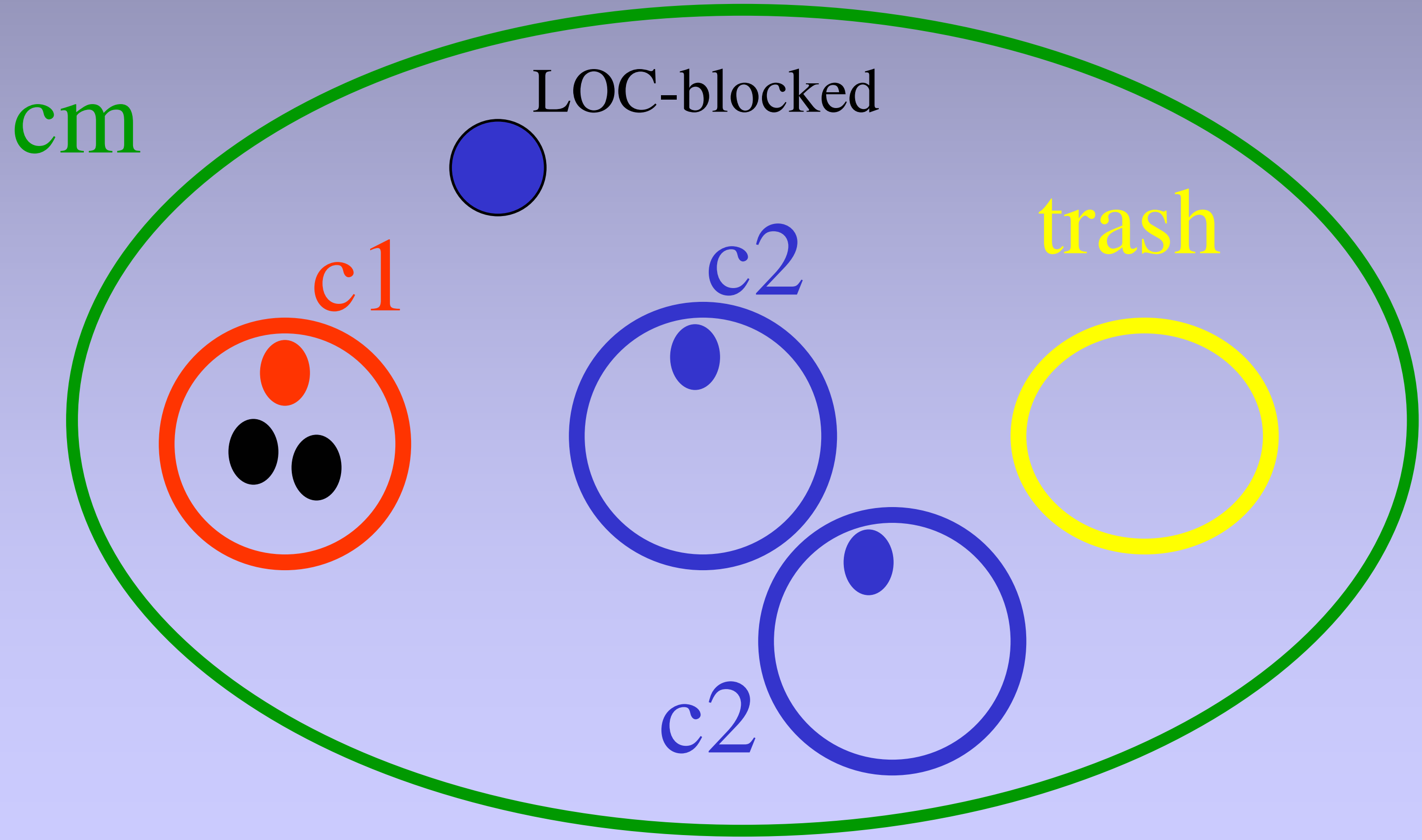
Zero-test I

- Suppose we need to test if c2 is empty



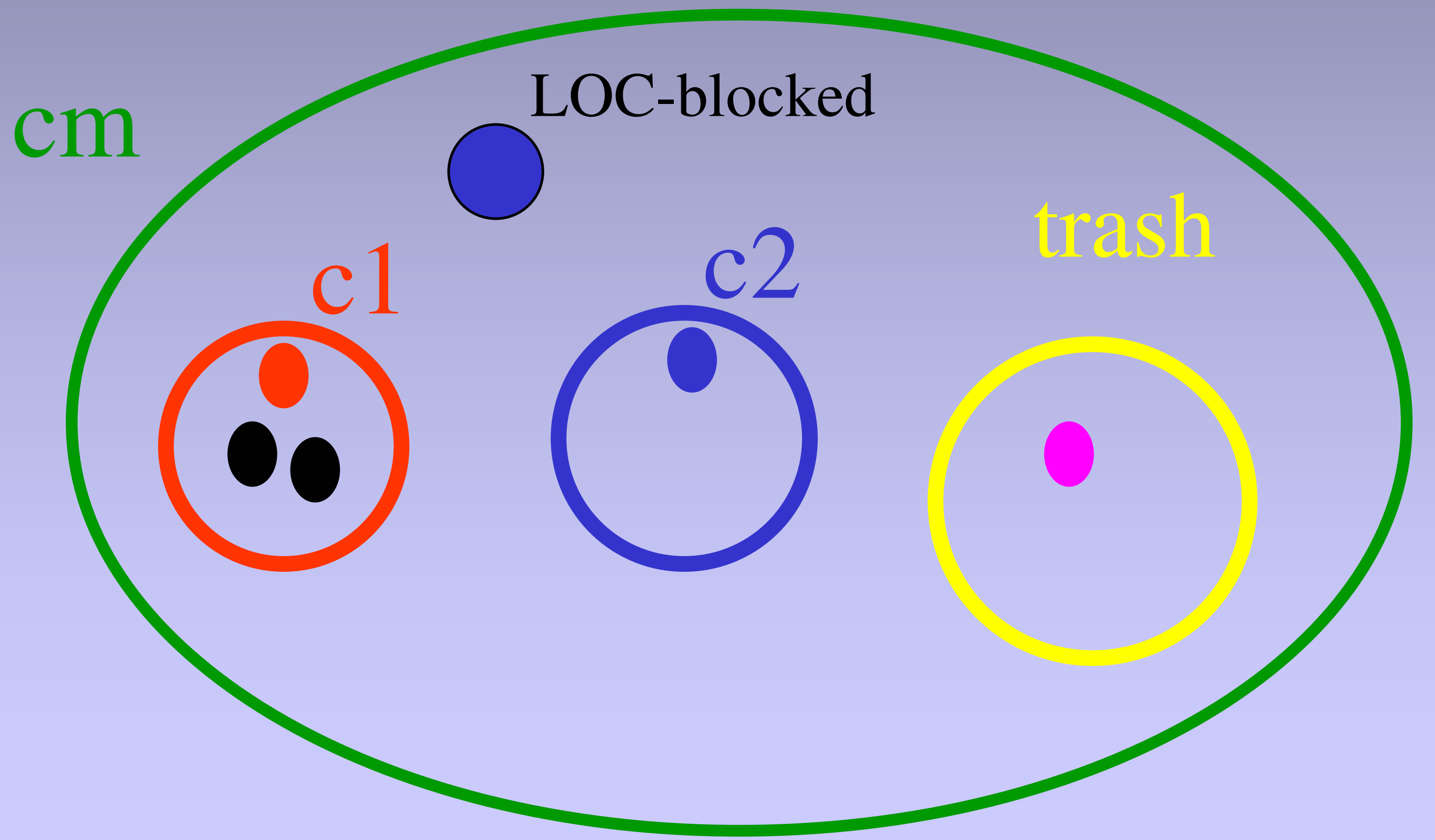
Zero-test II

- We block normal execution and clone c2



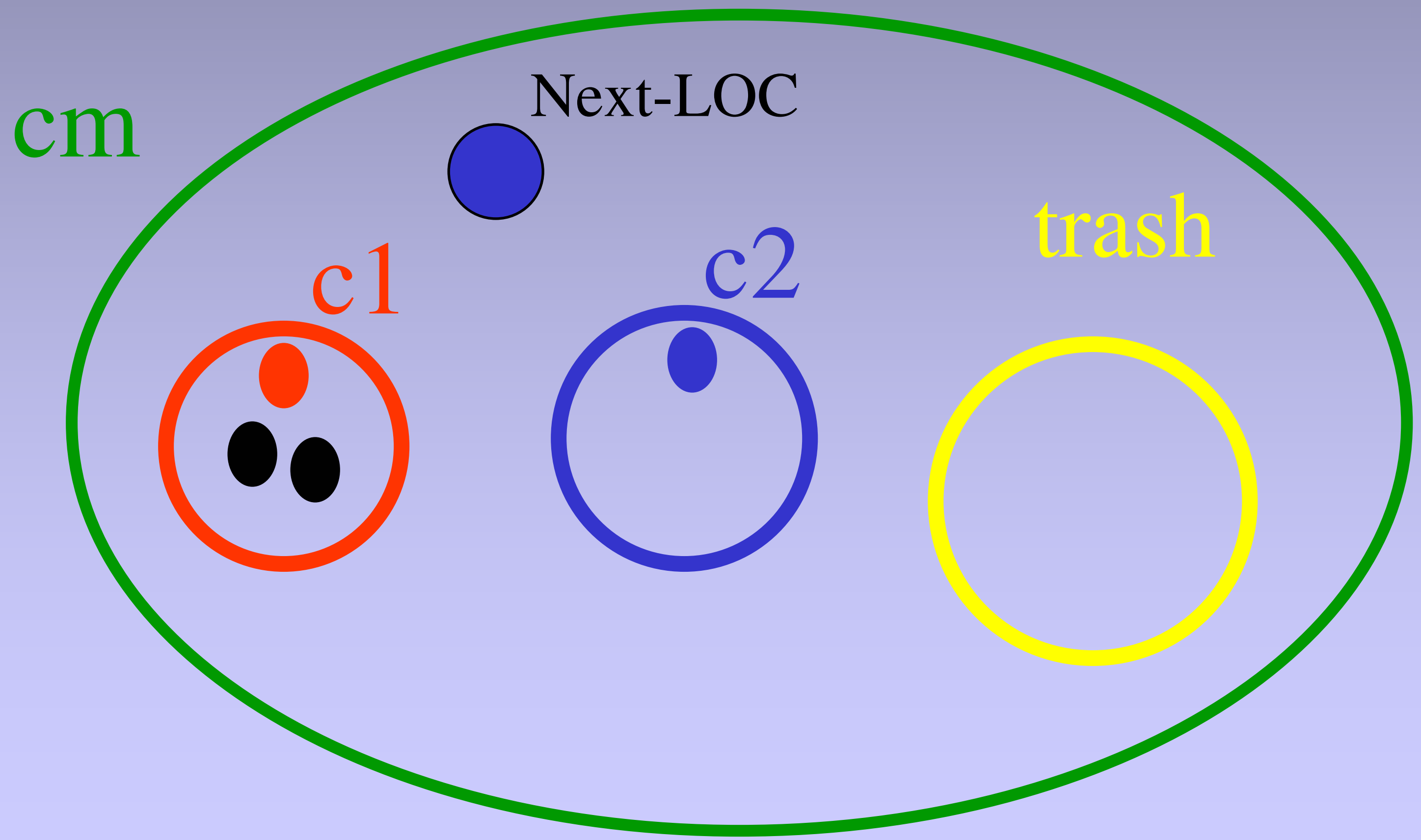
Zero-test III

- We then merge the copy with trash



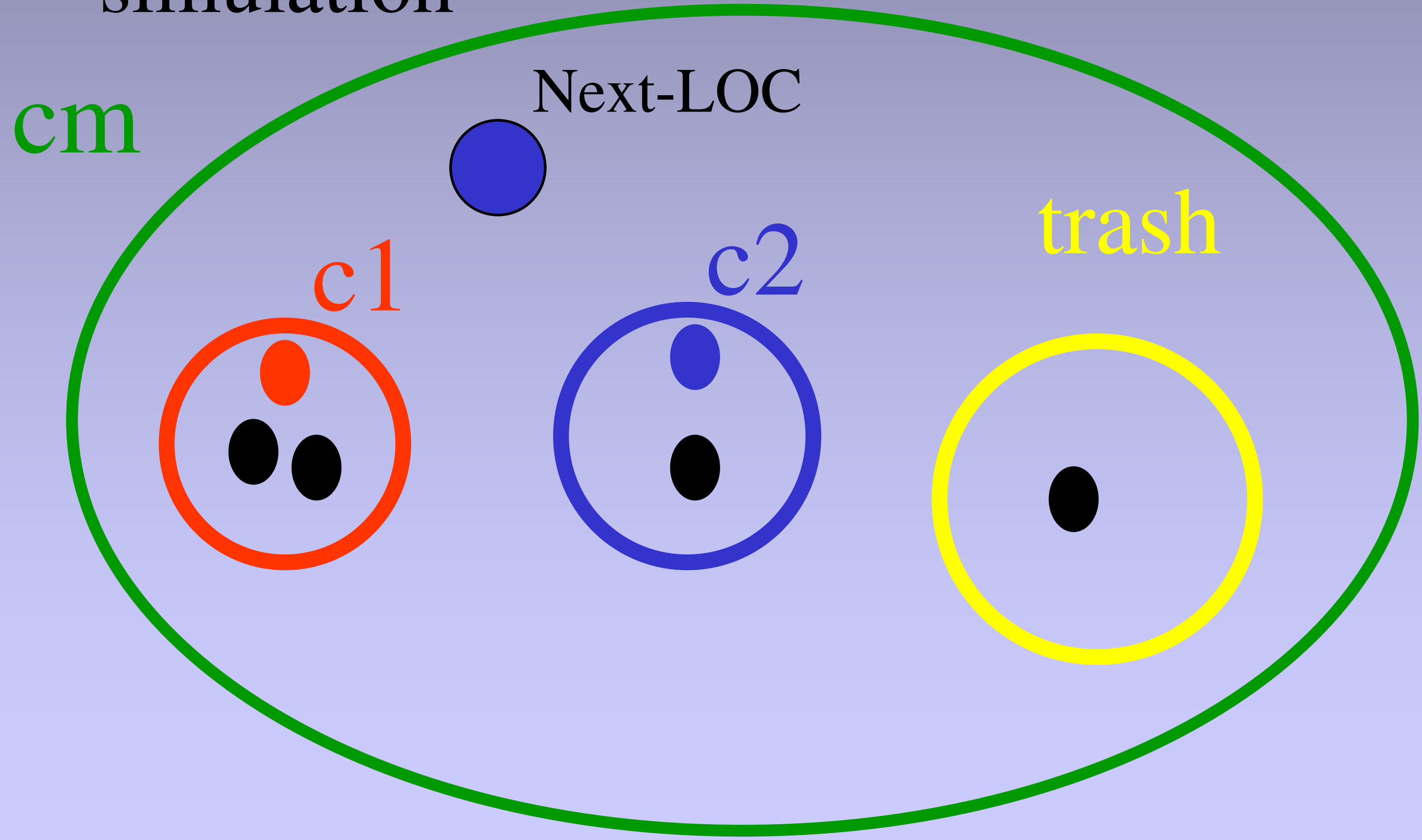
Zero-test IV

- Finally, we restart the normal execution



Remark

- If c2 is not zero, trash is not empty after the simulation



Property of the Encoding

- We may take wrong turns (we execute a zero-test on a non-zero counter)
- However, all executions in which configurations have empty trash membrane are good
- Thus, we can encode RP for counter machines (known to be undecidable) into RP for PBFC systems

Boundedness Problem (BP)

Definition of BP

- Assume a PB system with boundary, fusion and creation rules
- Fix a configurations C0
- REACH = the set of configurations reachable from C0

(BP) Is the set REACH finite?

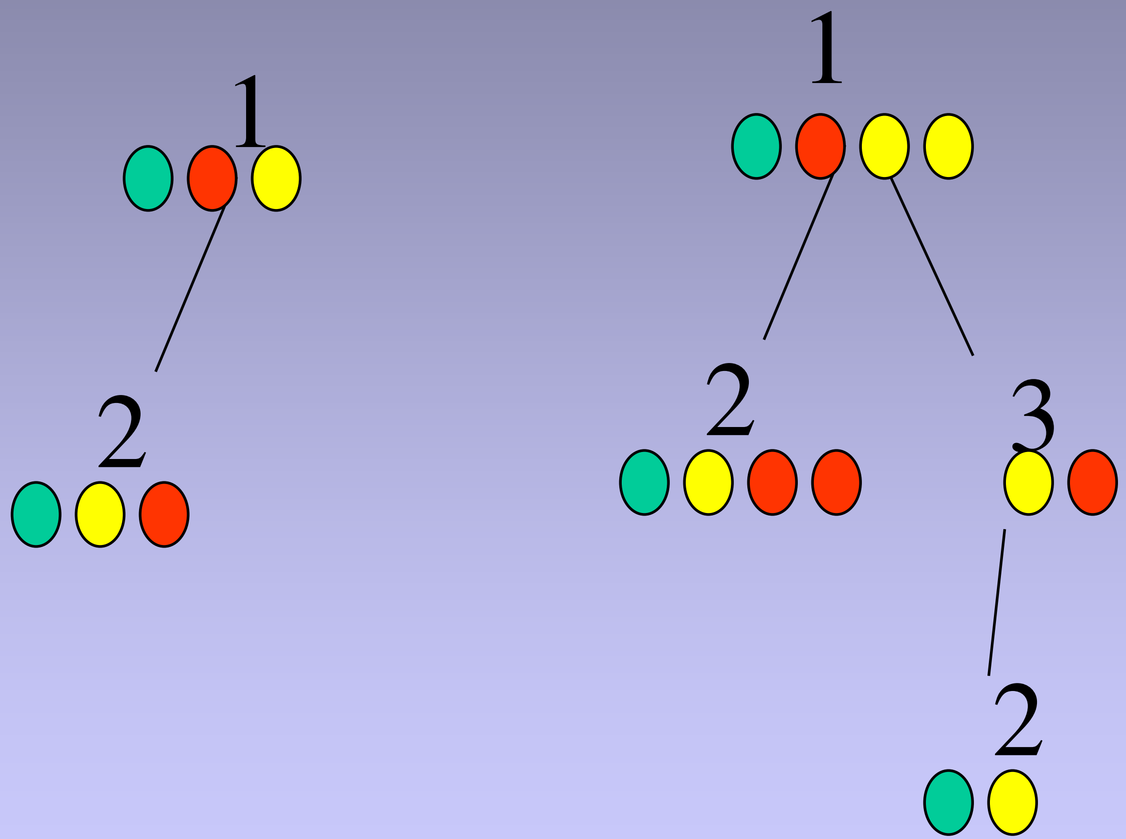
Our Result

BP is decidable for PBFC systems

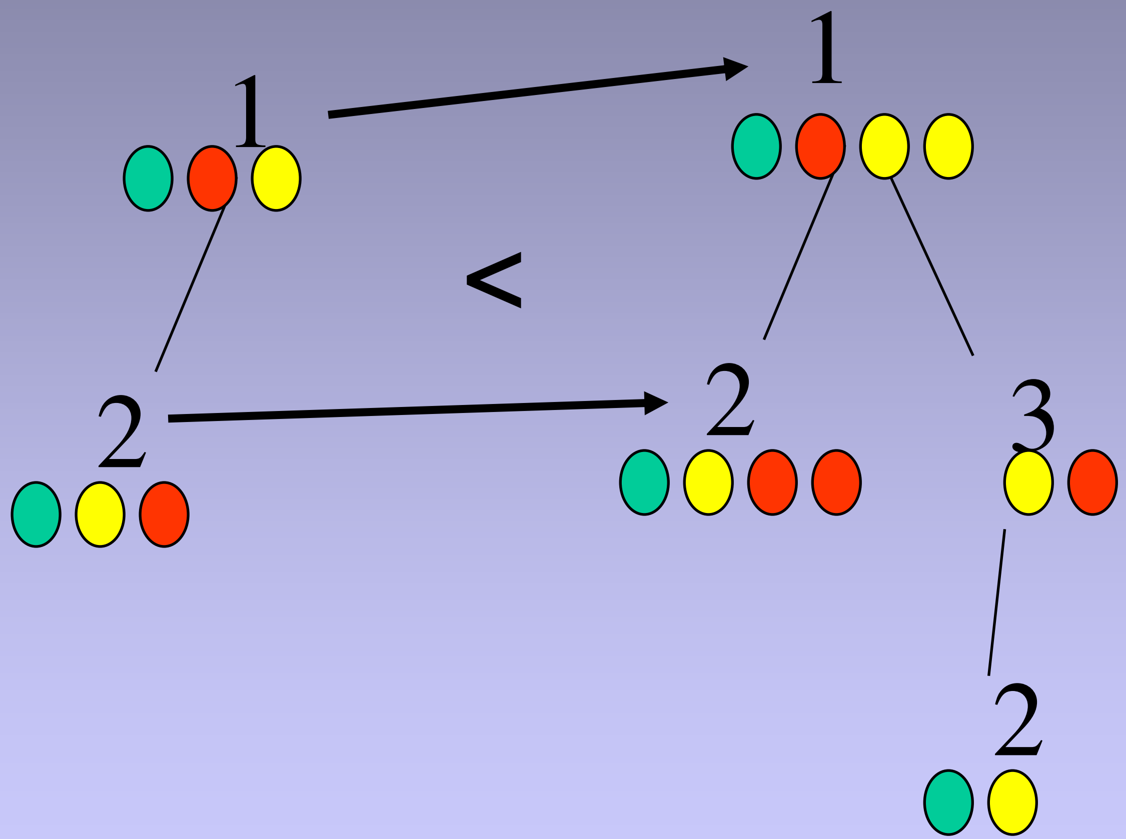
Proof

- Configurations are finite trees in which each node is labelled with a name and with a multiset of objects
- We first define an ordering < on configurations such that C<C' iff
  - C is mapped to a node in C' with the **same name** and **at least the same** objects as C' (inclusion of multisets)
  - The (injective) mapping preserves the father-son relationship of the immersion of C in C'

Ordering <



Ordering <



B-bounded Configurations

- Let B the depth (number of nested membranes) of the initial configuration
- The set REACH contains only trees whose depth is bounded by B
- *Notice that the width of trees in REACH is potentially unbounded*

Well-quasi ordering

- In the paper we show that < is a
  well-quasi ordering (wqo)
  for B-bounded configurations

- *Def wqo*:
  There are no infinite sequences of <-incomparable B-bounded configurations

Monotonicity

- PBFC systems are
  strictly monotonic
  with respect to <

- *Def:*
  If C → C’ and C < D, then there exists D’ such that C’<D’ and C’→ D’ where → is a firing step,

Well-structuredness

- A transition system that is monotonic w.r.t. a wqo defined on configuration is called well-structured [FS01]
- A forward reachability algorithm for checking boundedness for (strictly monotonic) well-structured transition systems is described in [FS01]

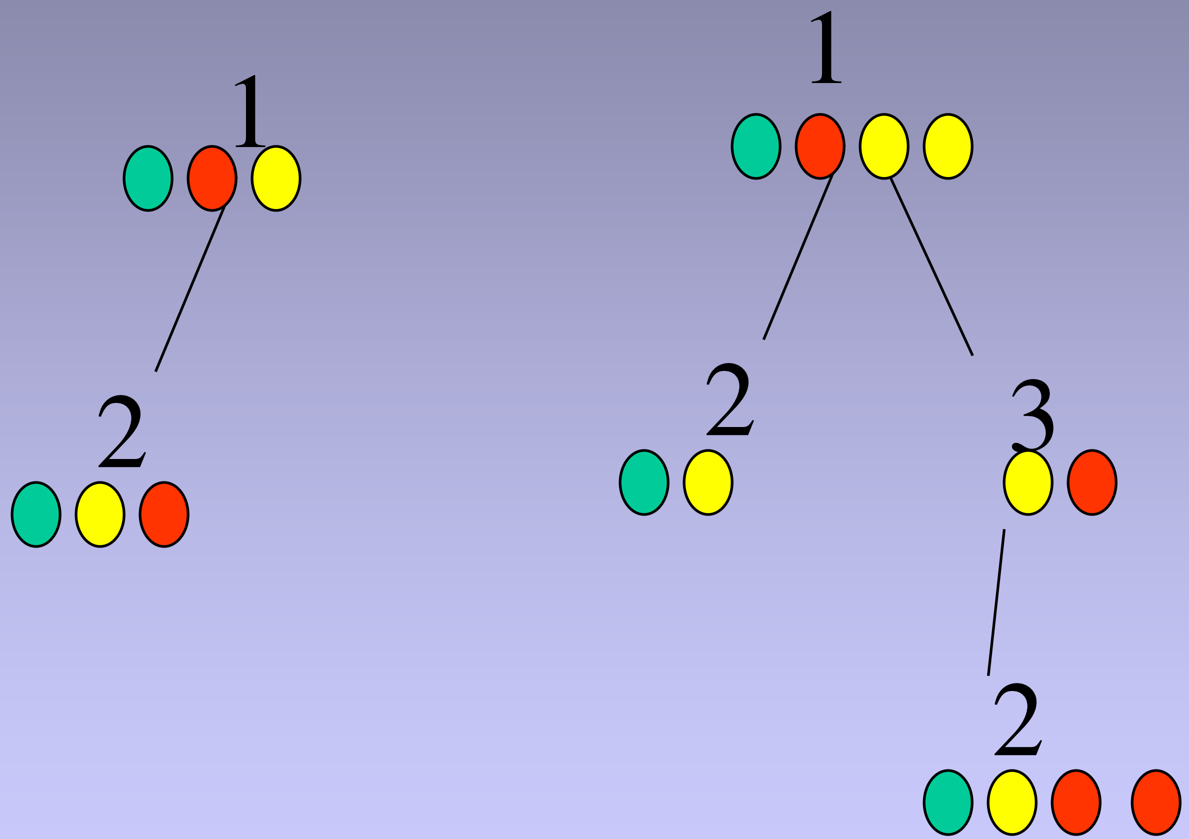
[FS01] Finkel-Schnoebelen
Well-structured Transition Systems everywhere TCS 2001

Coverability Problem (CP)

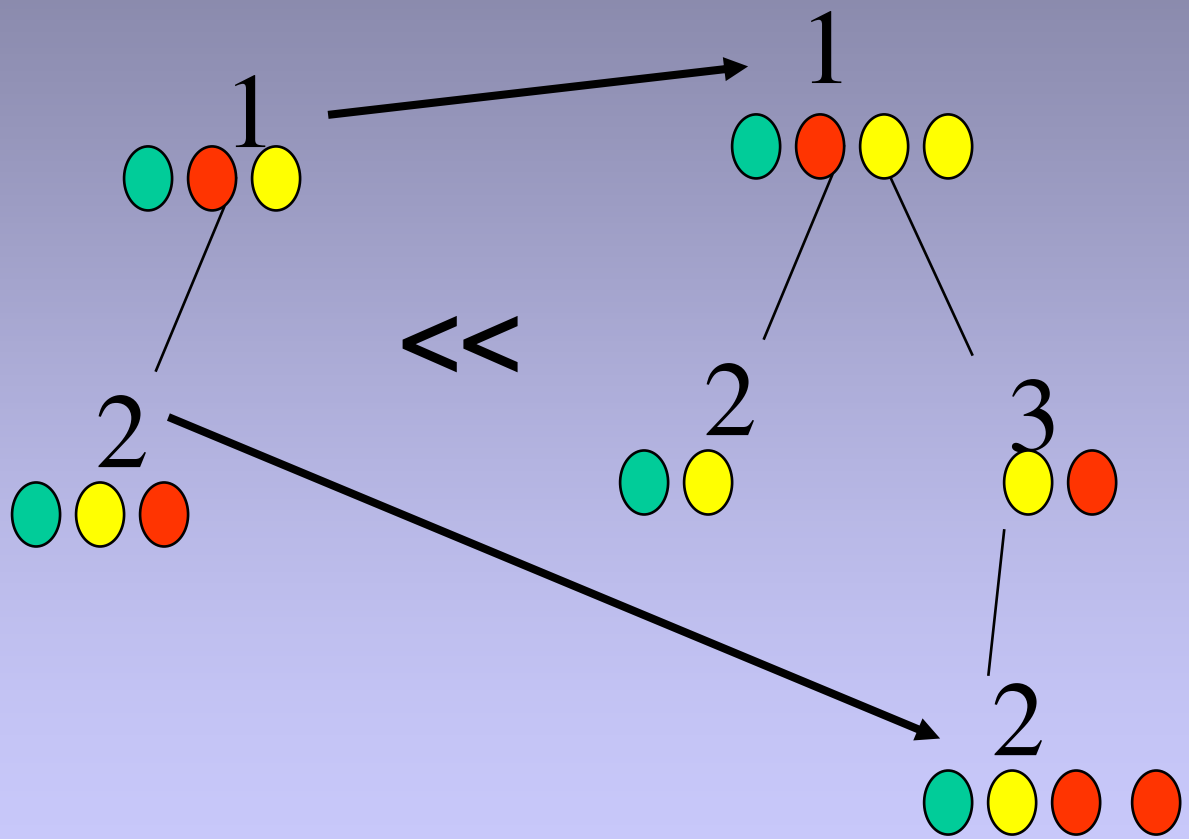
Kruskal Tree Embedding

- Let << be the tree embedding ordering in which labels are ordered by taking
  - equality for names and
  - inclusion for multisets of objects

Ordering << on configurations



Ordering << on configurations



Definition of CP

- Assume a PB system with boundary, fusion and creation rules

- Fix two configurations C0 and C1

(CP) Is there a configuration C2 reachable from C0 and such that C1 << C2 ?

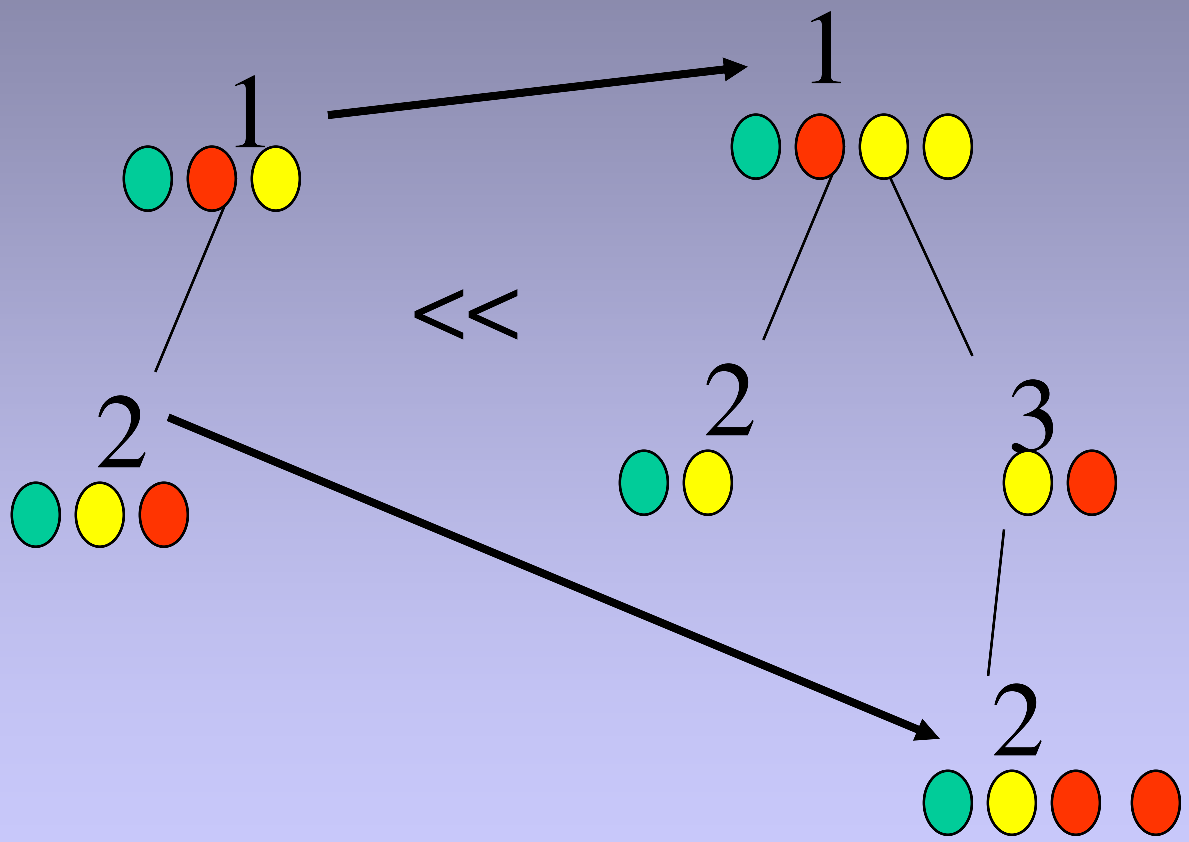
Our Result

Coverability is decidable for PBFC systems

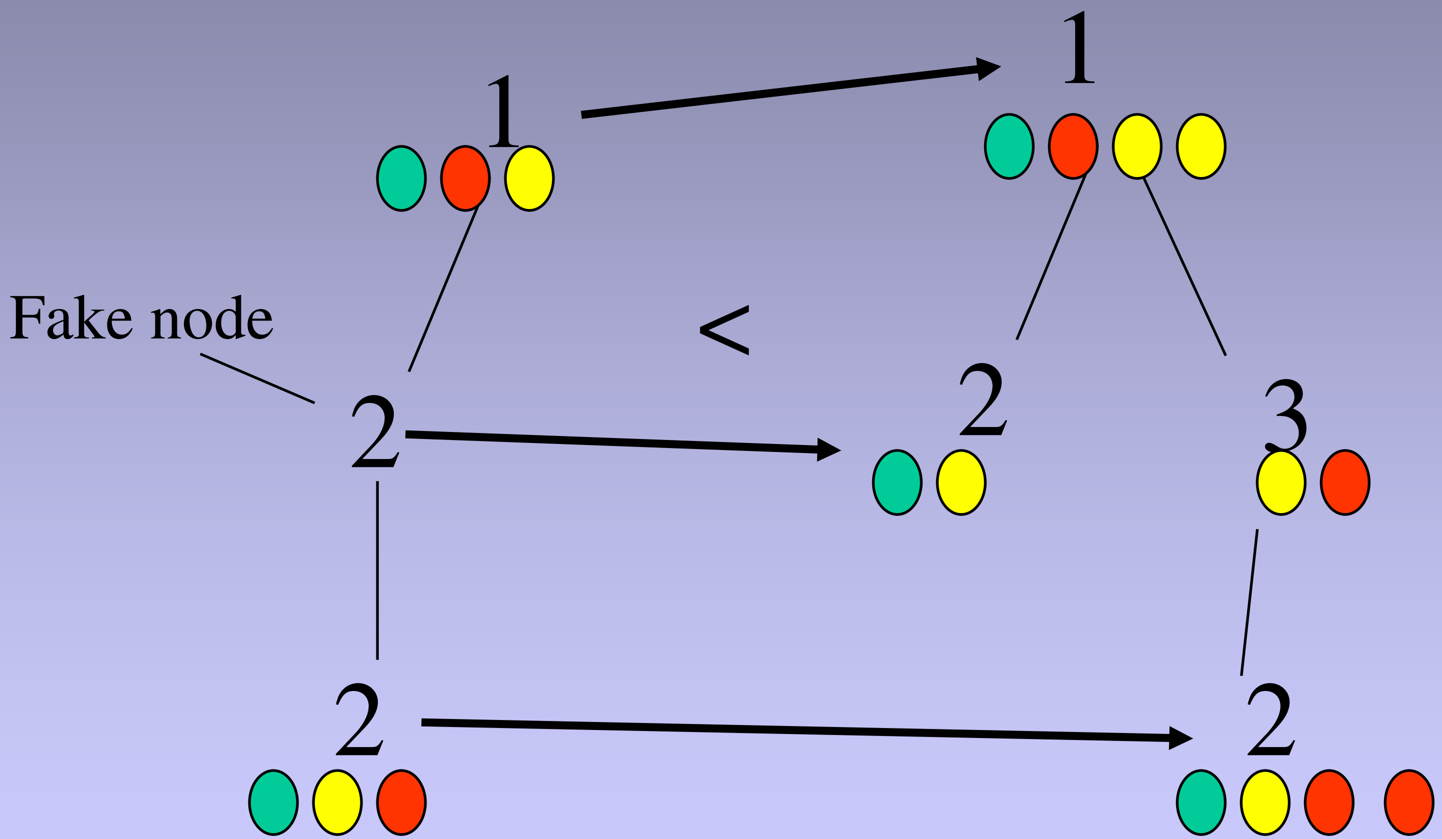
Proof

- We first solve the coverability problem with respect to the ordering < used for boundedness on B-bounded configurations
- We then reduce coverability for << to coverability for < by adding (when necessary) intermediate nodes (up to depth B)

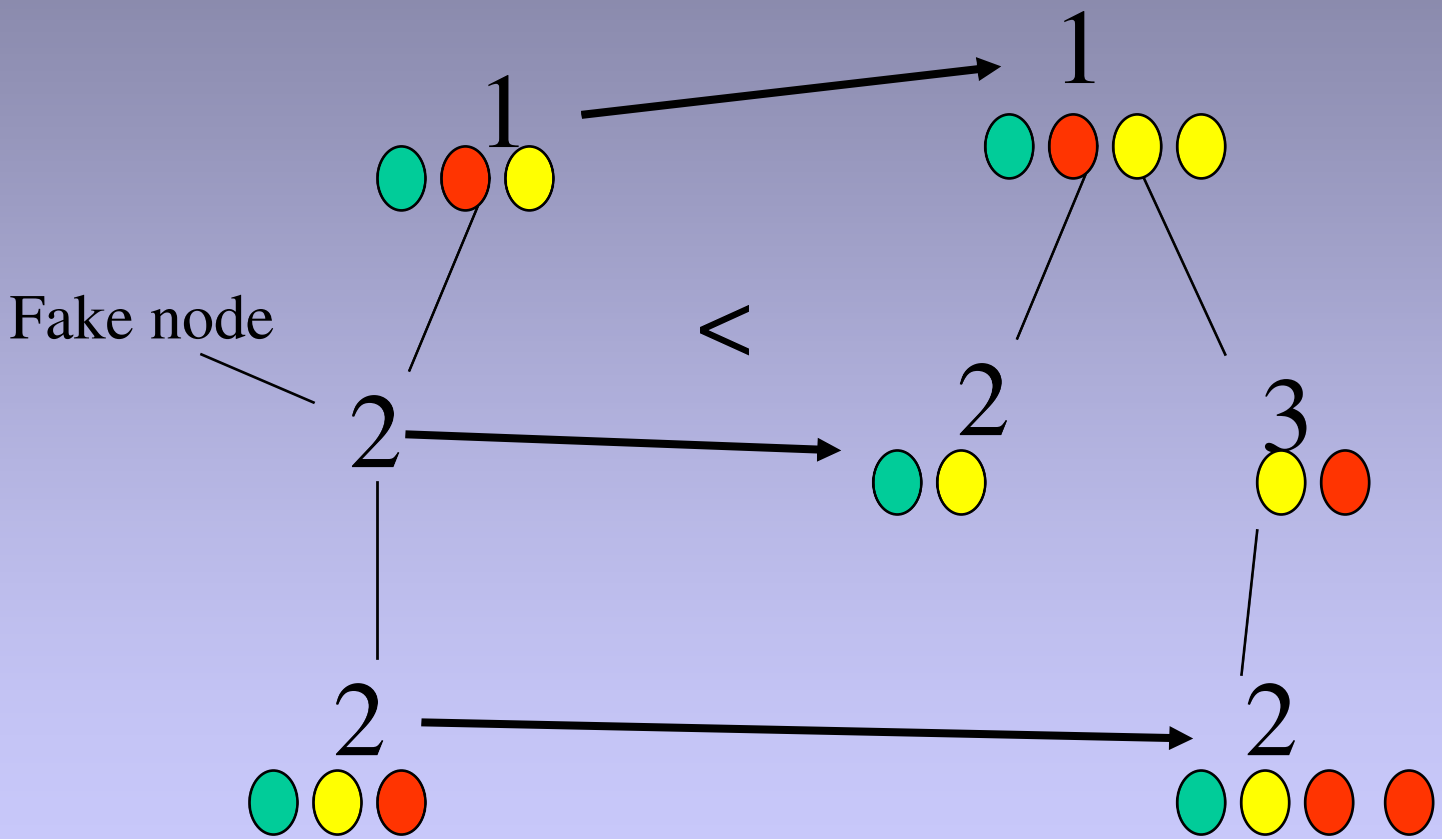
Encoding << using <



Encoding << using <



Encoding << using <



Remark: the depth is bounded by B: finitely many fake nodes!

Well-structuredness

- PBFC are strictly monotonic on <
- In the paper we show how to symbolically compute predecessors of upward closed sets of configurations
- Thus, we can apply a general result in [FS01] to define a symbolic backward search algorithm for checking coverability

[FS01] Finkel-Schnoebelen
***Well-structured Transition Systems everywhere* TCS 2001**

Conclusions

- We have defined a new model with biologically inspired operations
- We have shown that with interleaving semantics the model is in between Petri nets and Turing machines
  - Reachability is undecidable as for TM
  - Coverability and boundedness are decidable as for PN

Future work

- We plan to compare PBFC with (bio)concurrency models based on process algebra (e.g. bio/mobile ambients)

- We plan to study the expressiveness of combinations with other extensions of P-systems (e.g. dissolution and degradation)