

Capturing Membrane Computing by ASMs

Klaus-Dieter Schewe, Loredana Tec, Qing Wang

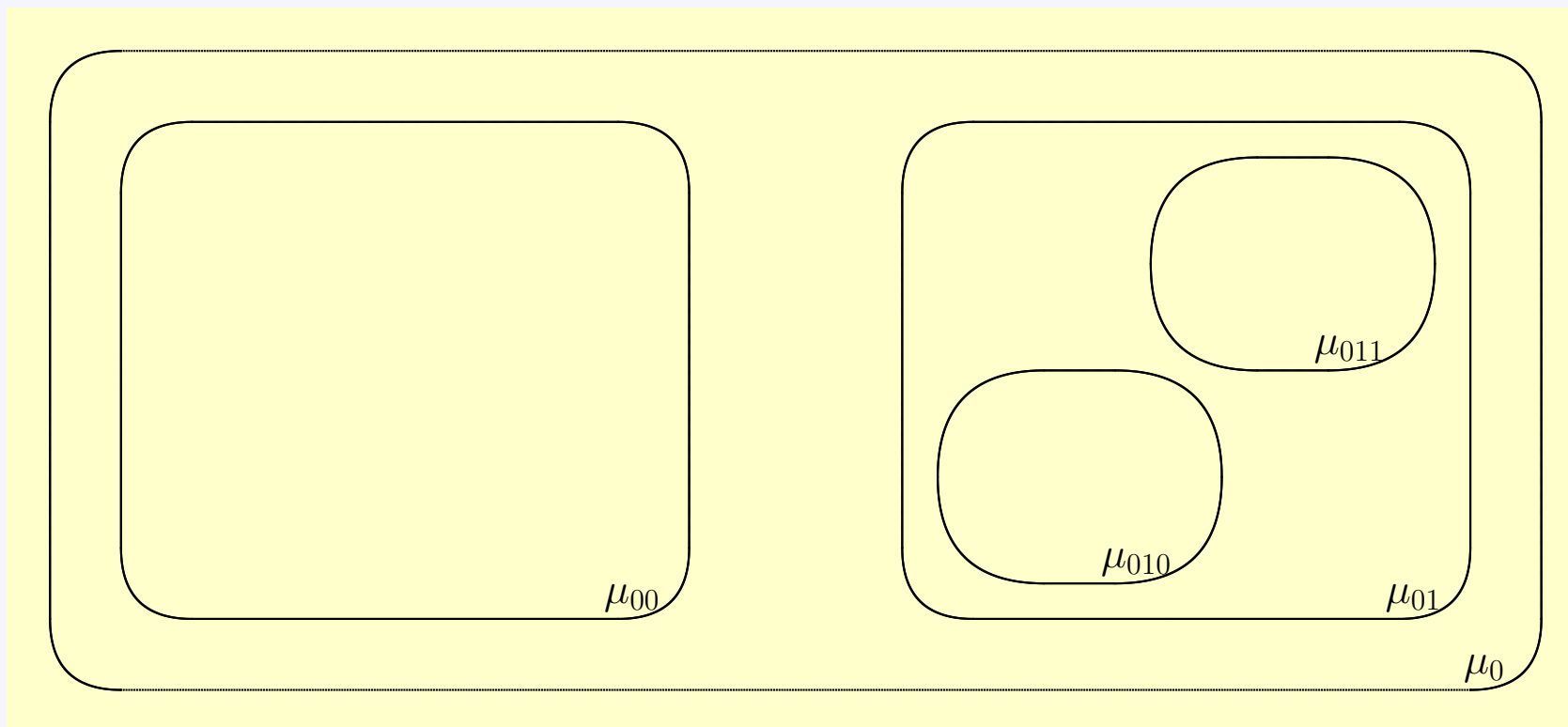
Linz, Austria & Canberra, Australia

kdschewe@acm.org

Membrane Structures

Membrane Structure: $ms = [\square[\square\square]]$

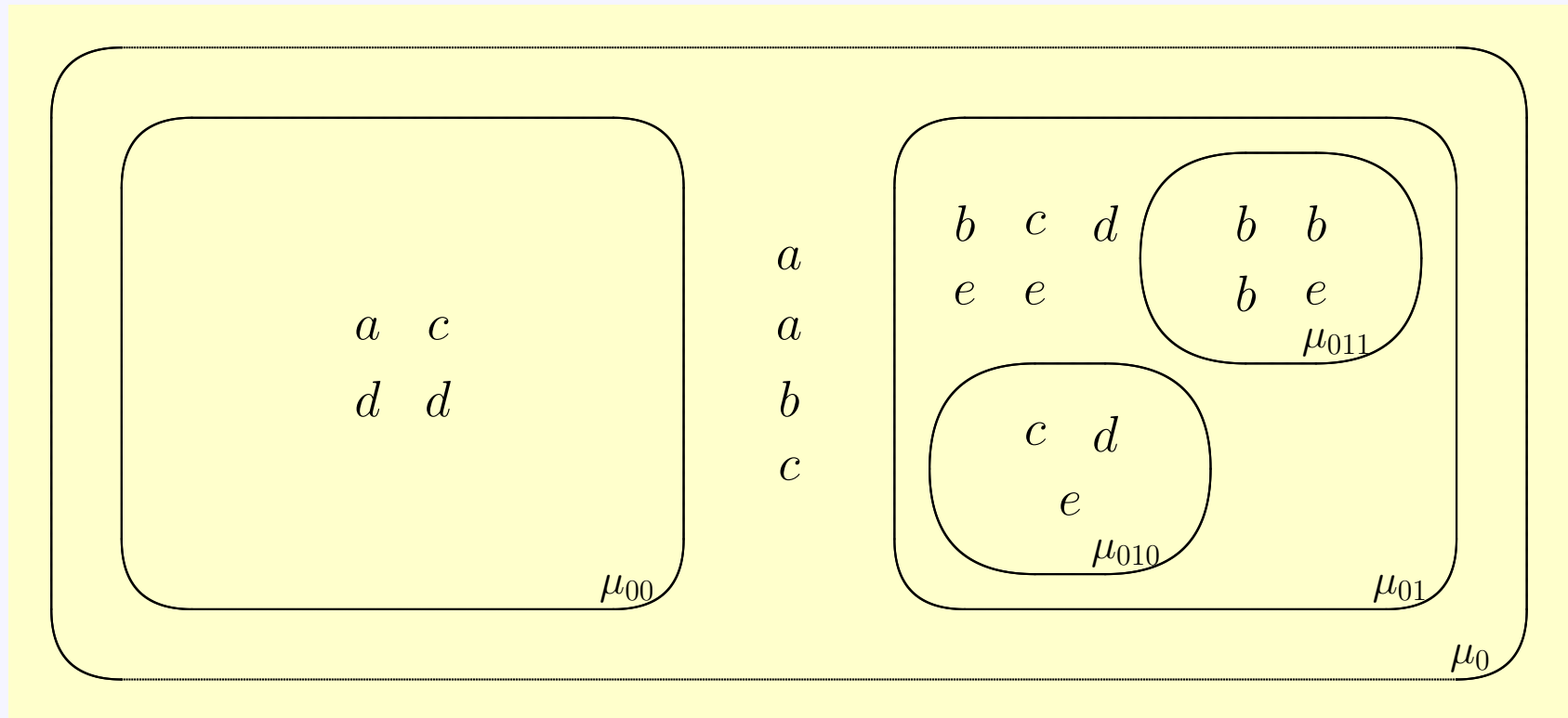
Set of membranes: $M_{ms} = \{\mu_0, \mu_{00}, \mu_{01}, \mu_{010}, \mu_{011}\}$



Super Cells

Set of Objects: $\mathcal{O} = \{a, b, c, d, e\}$

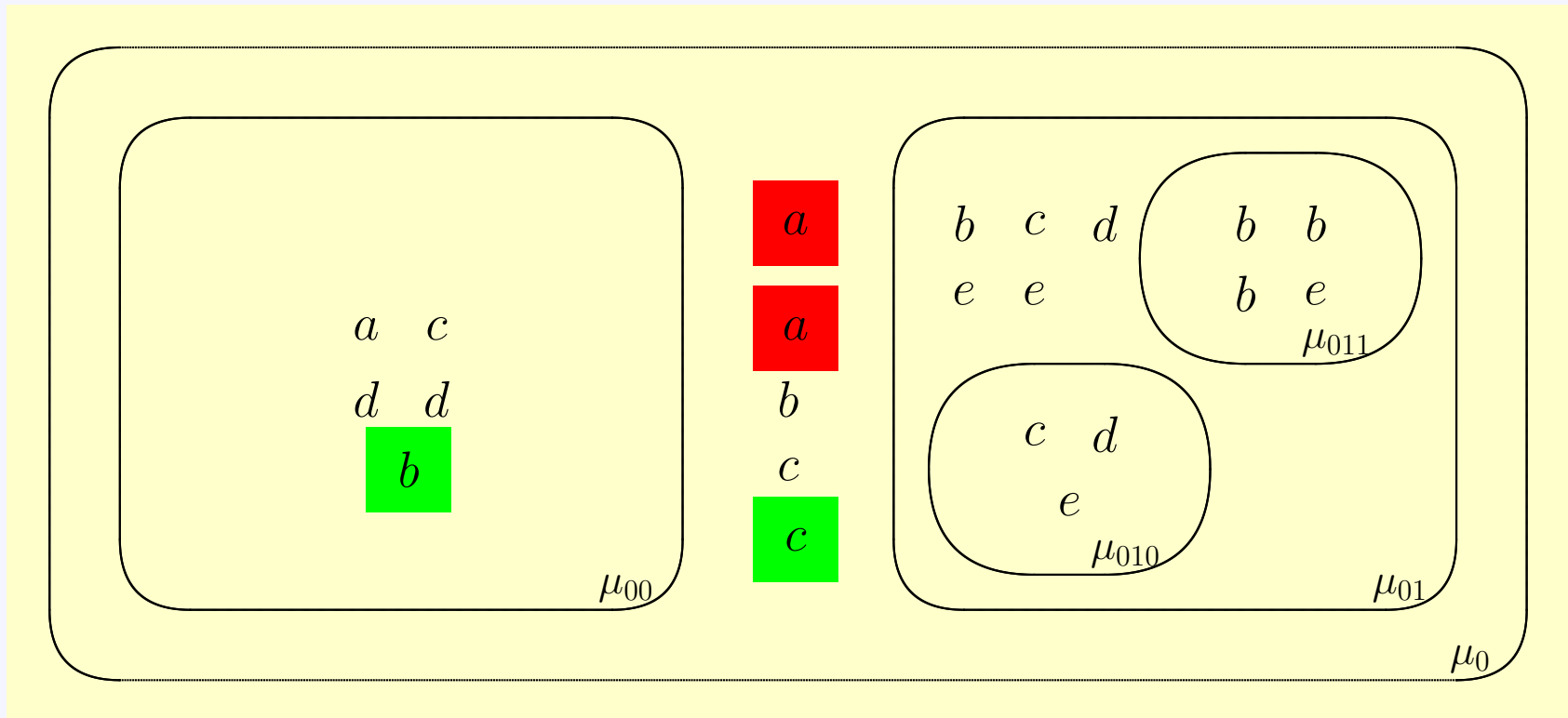
Association of objects: $o : M_{ms} \rightarrow \mathcal{O}^*$



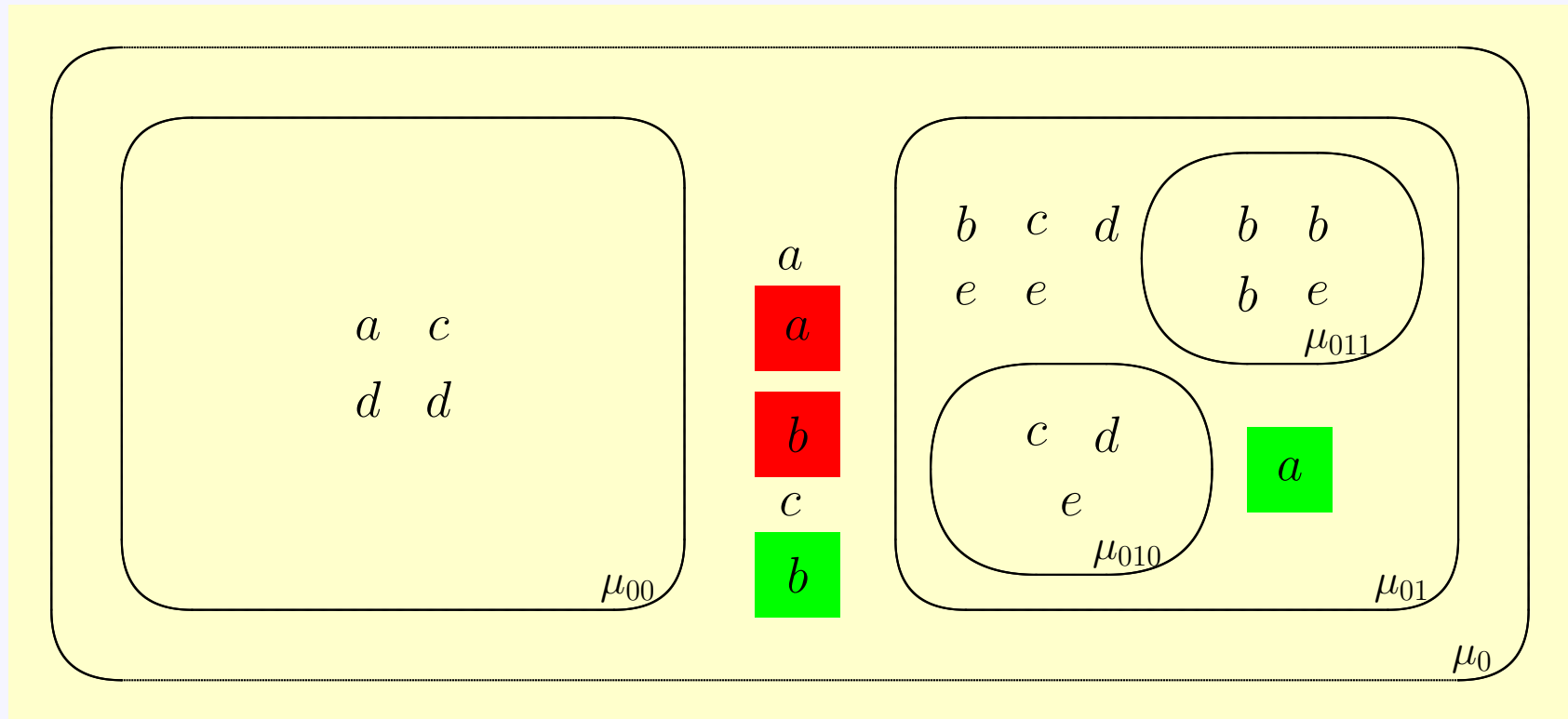
Rules / 1

$\hat{\mathcal{O}} = \mathcal{O} \times \{\text{here, out, in}_\mu\}$, $\delta \notin \mathcal{O}$, **P-System:** $(ms, \mathcal{O}, o, \mu_0, \varrho)$

Association of rules: $\varrho : M_{ms} \rightarrow \{\ell \rightarrow r \mid \ell \in \mathcal{O}^*, r \in \hat{\mathcal{O}}^* \cup \hat{\mathcal{O}}^* \delta\}$



Rules / 2



Rules are partially ordered by preference

A glimpse of practice: $abb \rightarrow cdd$ with

$a = COCl_2$ (phosgene), $b = NH_3$ (ammonia), $c = CO(NH_2)_2$ (urea), $d = HCl$ (hydro-chloride)

Extending the Scope

- We exploit unbounded parallelism and partial updates to capture P -systems by ASMs
 - However, it seems more realistic, if rules were executed in an asynchronous way (\rightarrow exploit concurrent ASMs)
- P -systems change their states, which are super cells, but rules remain fixed
 - It seems more realistic, if rules were also subject to change (\rightarrow exploit reflective ASMs)
- Objects in P -systems are elements of an alphabet \mathcal{O}
 - It seems more realistic, if objects had a deep structure (\rightarrow exploit Tarski structures underlying ASMs)
- P -systems are to mimick real biological systems
 - It seems more realistic to consider also continuous behaviour (\rightarrow exploit hybrid ASMs)