Membrane Systems in Algebraic Biology: From a Toy to a Tool

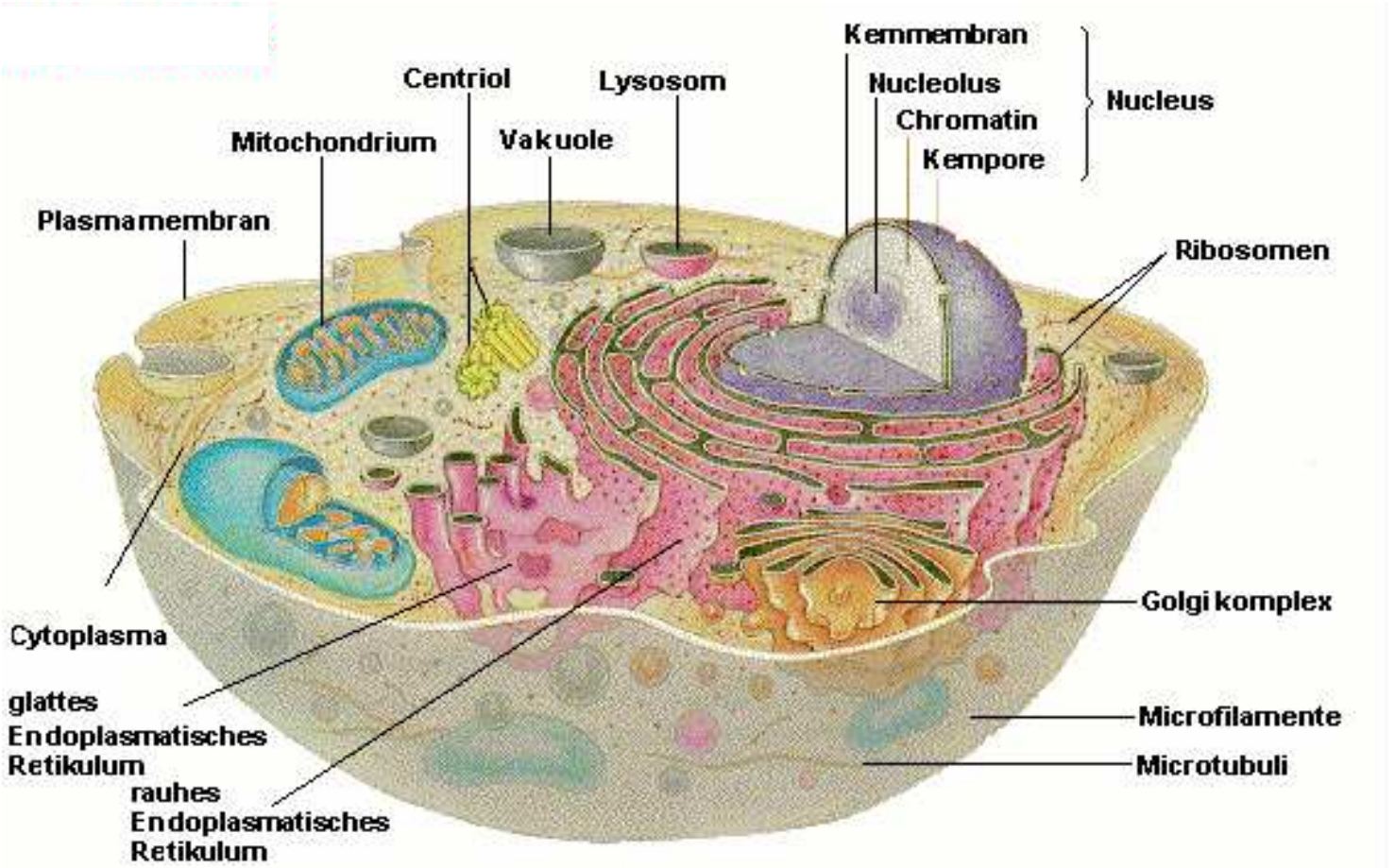
Thomas Hinze

Friedrich Schiller University Jena
School of Biology and Pharmacy
Department of Bioinformatics

thomas.hinze@uni-jena.de

January 29, 2009

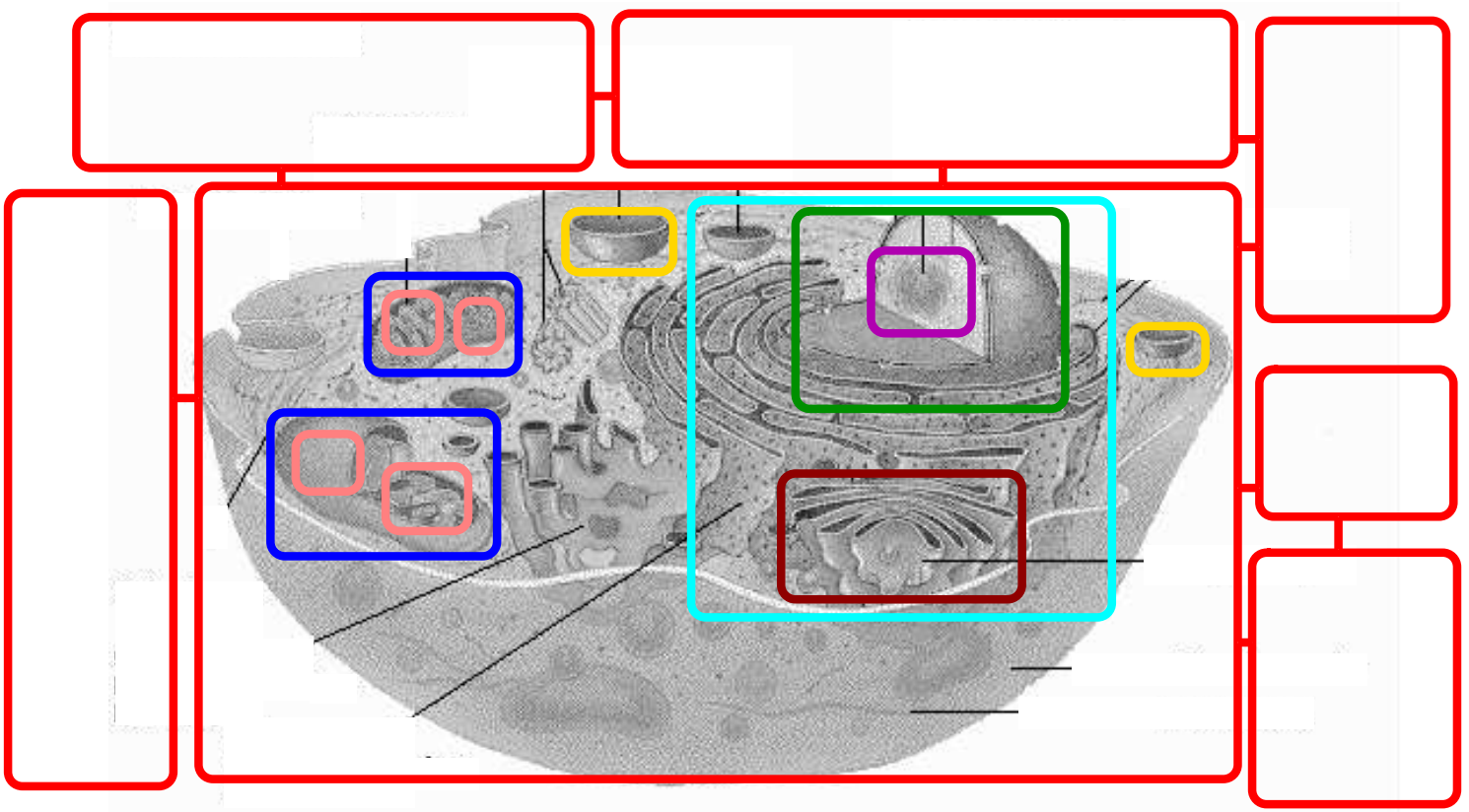
Membrane Systems: Inspired by Cells and Tissues



www.zum.de

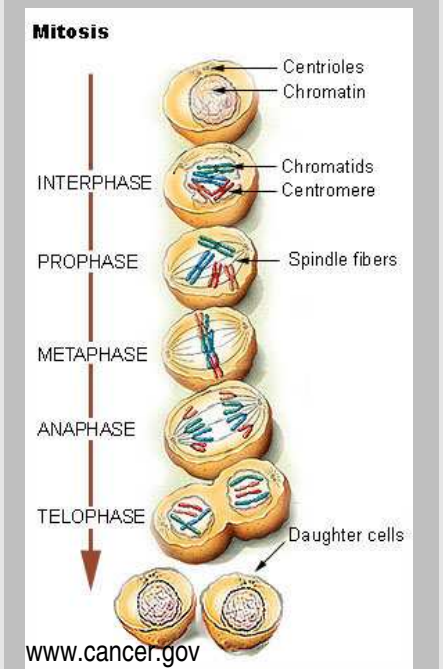
$\Longrightarrow$ Capturing specialties of intra- and intercellular processes

(I) Nested Compartments Delimited by Permeable Membranes

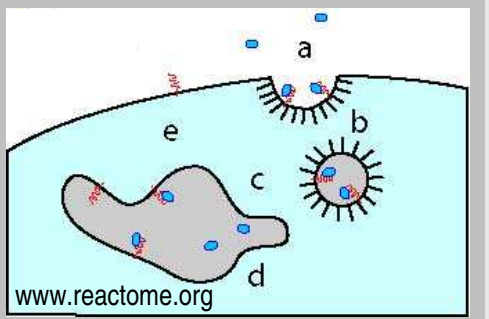


$\Longrightarrow$ Spatial regions wherein chemical reactions can occur

(II) Dynamics in Compartmental Cell Structure



division

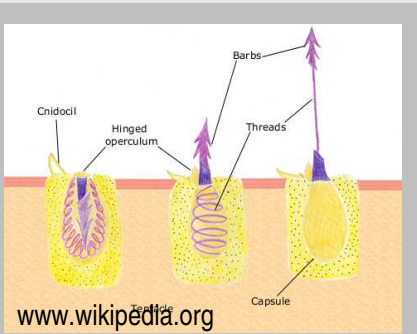


endocytosis

merge

exocytosis

separation



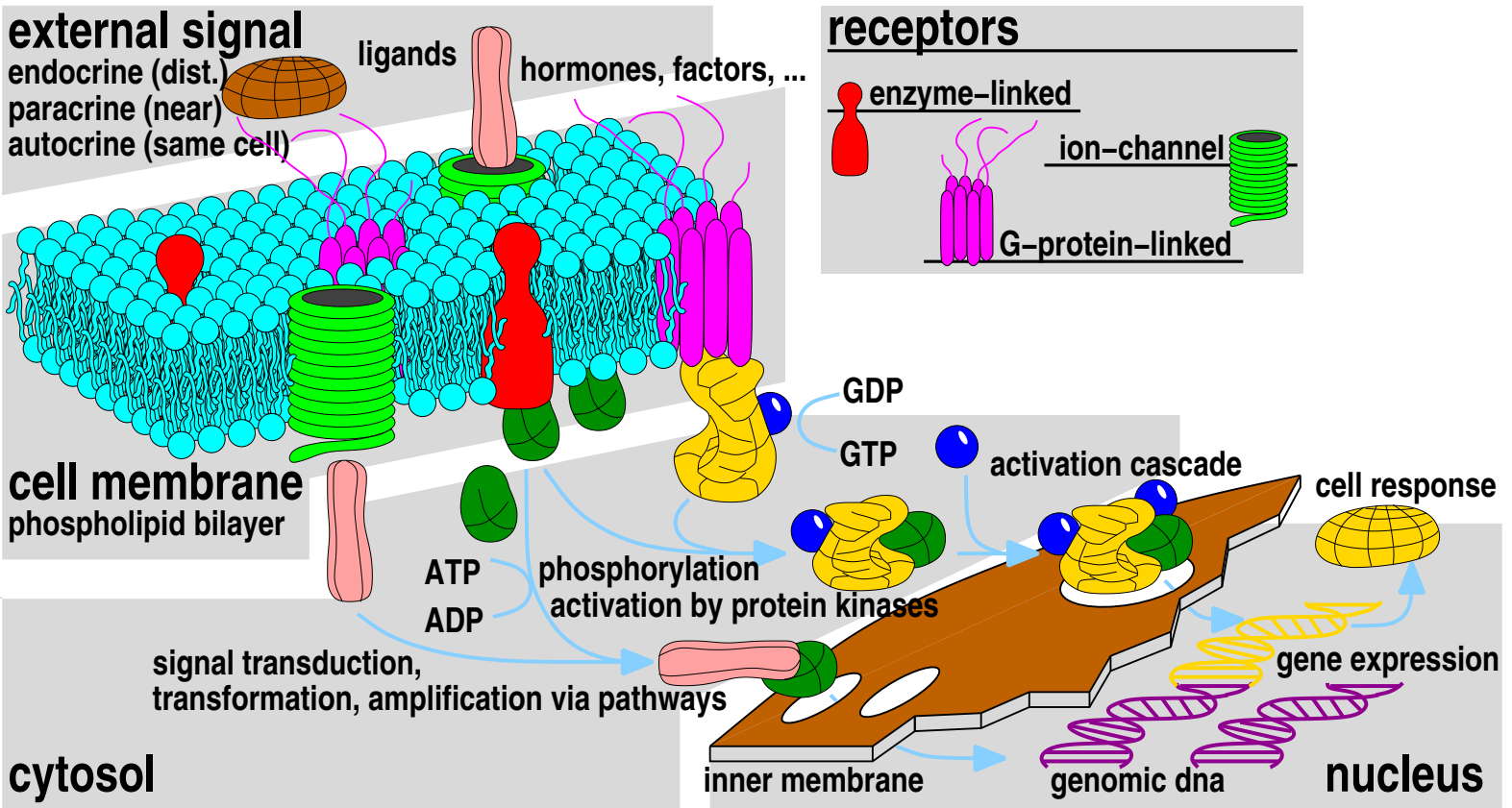
gemmation

creation

dissolution

$\Longrightarrow$ Plasticity initiated by dedicated reaction networks

(III) Complex Polymeric Biomolecules in Low Concentrations

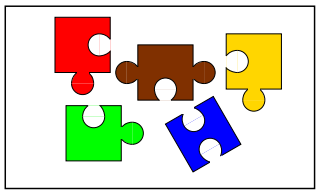
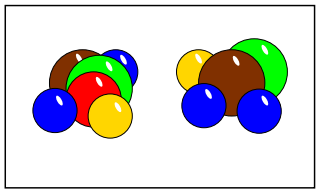
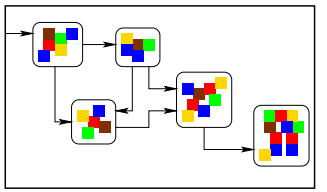
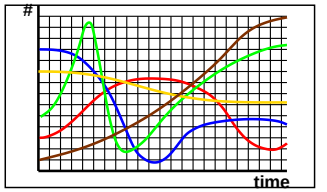


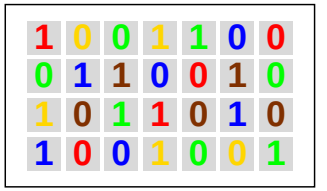
$\Longrightarrow$ Reaction pathways forming specific biomolecules

Outline

Membrane Systems in Algebraic Biology: From a Toy to a Tool

1. **A Primer: Reaction systems**
   composed of discrete entities
2. **Membrane systems:**
   Some introductory examples
3. **Features and varieties:**
   Classification and properties
4. Π_{CSN}**: A modelling framework**
   for cell signalling networks
5. **Bio-Applications:**
   Appetizers for systems biologists
6. **Membrane systems as computing devices**
7. **The P page:** An online repository and more
8. **Quo vadis:** Concluding remarks and outlook

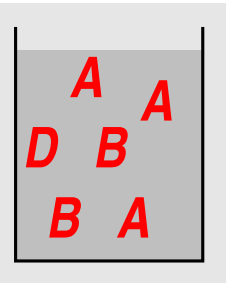







Multiset: Molecular Configuration within a Membrane

Example

$$\begin{aligned} \mathcal{L} &= \{(A,3),(B,2),(C,0),(D,1)\} \\ \text{supp}(\mathcal{L}) &= \{A,B,D\} \\ \text{card}(\mathcal{L}) &= 6 \end{aligned}$$



contents of a toy membrane

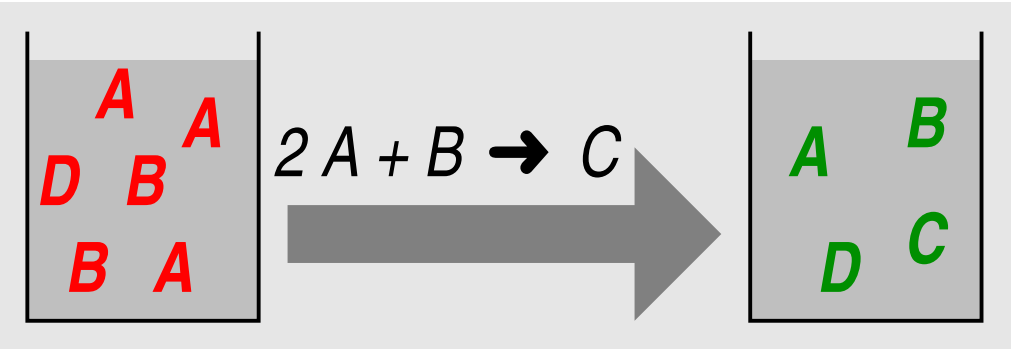
Definitions

Multiset: Let F be a set. A multiset over F is a mapping $\mathcal{F} : F \longrightarrow \mathbb{N} \cup \{\infty\}$ that specifies the multiplicity of each element $a \in F$.

Support: Let $\mathcal{F} : F \longrightarrow \mathbb{N} \cup \{\infty\}$ be a multiset. A set $S \subseteq F$ is called $\text{supp}(\mathcal{F})$ iff $S = \{s \in F \mid \mathcal{F}(s) > 0\}$.

Cardinality: $\text{card}(\mathcal{F}) := \sum_{a \in F} \mathcal{F}(a)$

Term Rewriting: Employ a Reaction by Set Operations



Example

$\{(A,3),(B,2),(C,0),(D,1)\} \ominus \{(A,2),(B,1)\} \uplus \{(C,1)\} = \{(A,1),(B,1),(C,1),(D,1)\}$

Multiset operations

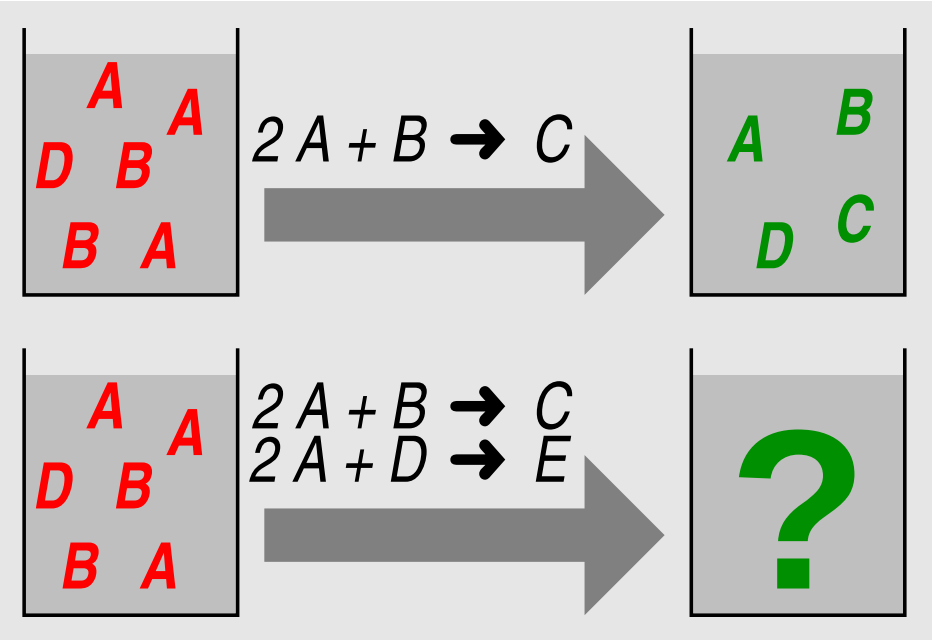
Difference: $\mathcal{F} \ominus \mathcal{G} := \{(a, \max(\mathcal{F}(a) - \mathcal{G}(a), 0)) \mid a \in F \setminus G\}$

Sum: $\mathcal{F} \uplus \mathcal{G} := \{(a, \mathcal{F}(a) + \mathcal{G}(a)) \; a \in F \cup G\}$

Union: $\mathcal{F} \cup \mathcal{G} := \{(a, \max(\mathcal{F}(a), \mathcal{G}(a))) \mid a \in F \cup G\}$

Intersection: $\mathcal{F} \cap \mathcal{G} := \{(a, \min(\mathcal{F}(a), \mathcal{G}(a))) \mid a \in F \cap G\}$

Coping with Conflicts: Introducing Process Control



Possible strategies to decide among satisfied reactions

Nondeterminism: Maximal parallel enumeration of all potential scenarios

Prioritisation of reactions: Determinisation by a predefined order for applicability of reactions

Stochasticity: Randomly select a satisfied reaction

Evolution over Time

Time-discrete iteration scheme

- Starting from an initial molecular configuration $\mathcal{L}_0$
- Iterative term rewriting for transition(s) $\mathcal{L}_t \rightarrow \mathcal{L}_{t+1}$. Each iteration corresponds to a discrete period in time $\Delta\tau$
- Within each iteration turn, applicable reactions are figured out and subsequentially employed once or several times (e.g. kinetic function $\mathrm{f} : \mathcal{L} \rightarrow \mathbb{N}$ in concert with discretised kinetic laws)
- Obtaining a derivation tree that lists sequences of molecular configurations as nodes

Considered aspects

- Suitability for small amounts of reacting particles (e.g. cell signalling)
- Compliance with mass conservance for undersatisfied reaction scenarios

First Example: A Single Membrane System

$\Pi_{\mathrm{PR}} = (V, T, [_1]_1, \mathcal{L}_0, R)$

V ..... system alphabet
$T \subseteq V$ ..... terminal alphabet
$[_1]_1$ ..... compartmental structure
$\mathcal{L}_0 \subset V \times (\mathbb{N} \cup \{\infty\})$ ..... multiset for initial configuration
$R = \{r_1, \ldots, r_k\}$ ..... set of reaction rules

Each reaction rule r_i consists of two multisets (reactants $\mathcal{E}_i$, products $\mathcal{P}_i$) such that

$$r_i = (\{(A_1, a_1), \ldots, (A_h, a_h)\}, \{(B_1, b_1), \ldots, (B_v, b_v)\}).$$

We write in chemical denotation:

$$r_i : a_1 A_1 + \ldots + a_h A_h \longrightarrow b_1 B_1 + \ldots + b_v B_v$$

$\Longrightarrow$ Index i specifies priority of r_i: $r_1 > r_2 > \ldots > r_k$.

First Example: A Single Membrane System

$\Pi_{\mathrm{PR}} = (V, T, [_1]_1, \mathcal{L}_0, R)$

V ..... system alphabet
$T \subseteq V$ ..... terminal alphabet
$[_1]_1$ ..... compartmental structure
$\mathcal{L}_0 \subset V \times (\mathbb{N} \cup \{\infty\})$ ..... multiset for initial configuration
$R = \{r_1, \ldots, r_k\}$ ..... set of reaction rules

Each reaction rule r_i consists of two multisets (reactants $\mathcal{E}_i$, products $\mathcal{P}_i$) such that

$$r_i = (\{(A_1, a_1), \ldots, (A_h, a_h)\}, \{(B_1, b_1), \ldots, (B_v, b_v)\}).$$

We write in chemical denotation:

$$r_i : a_1 A_1 + \ldots + a_h A_h \longrightarrow b_1 B_1 + \ldots + b_v B_v$$

$\Longrightarrow$ Index i specifies priority of r_i: $r_1 > r_2 > \ldots > r_k$.

Dynamical Behaviour of Π_{PR}

Iteration scheme for configuration update incrementing discrete time points $t \in \mathbb{N}$

$$\begin{aligned}\mathcal{L}_{t+1} = \{ (a, \alpha_{a,k}) \mid & \forall a \in V \\ & \wedge\ \alpha_{a,0} = \mathrm{card}\,(\mathcal{L}_t \cap \{(a,\infty)\}) \\ & \wedge\ \beta_{a,i} = \mathrm{card}\,(\mathcal{E}_i \cap \{(a,\infty)\}) \\ & \wedge\ \gamma_{a,i} = \mathrm{card}\,(\mathcal{P}_i \cap \{(a,\infty)\}) \\ & \wedge\ \alpha_{a,i} = \begin{cases} \alpha_{a,i-1} + \mathrm{f}_i \cdot \gamma_{a,i} - \mathrm{f}_i \cdot \beta_{a,i} & \text{iff } \forall a \in V : \alpha_{a,i-1} \geq \mathrm{f}_i \cdot \beta_{a,i} \\ \alpha_{a,i-1} & \text{else} \end{cases} \\ & \wedge\ i \in \{1, \ldots, k\} \}\end{aligned}$$

System output (distinction empty/nonempty configuration):

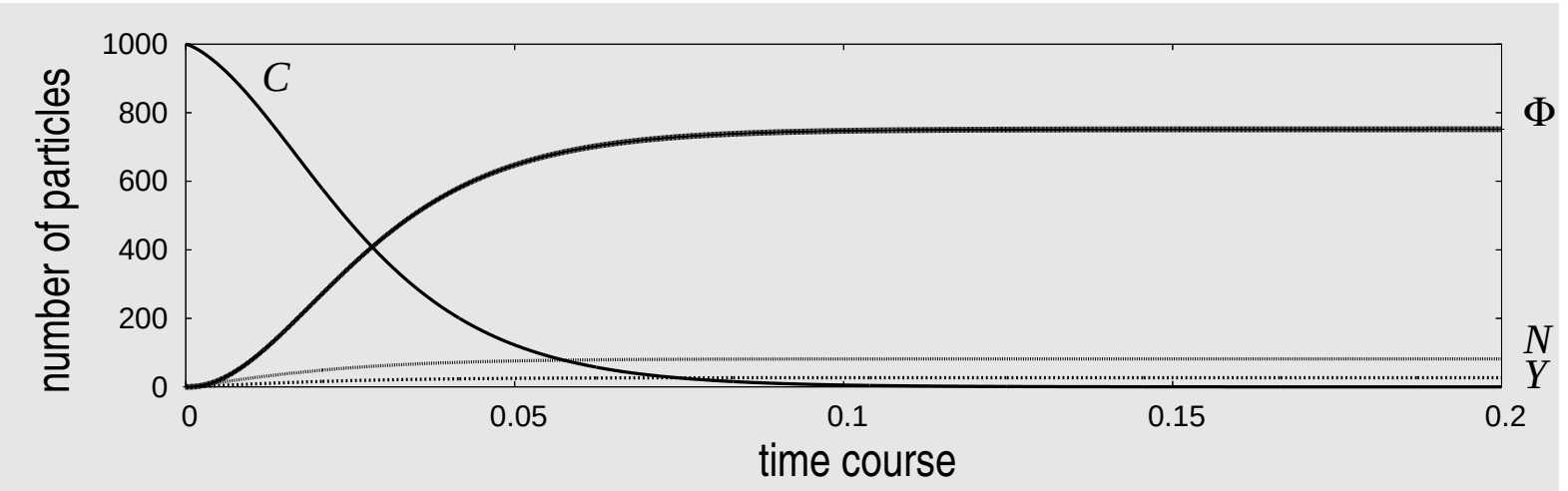
$$\mathrm{supp}\left(\biguplus_{t=0}^{\infty} (\mathcal{L}_t \cap \{(w,\infty) \mid w \in T\})\right) \subseteq T$$

Simulation Study of a Concrete Toy System Π_{PR}

$$\Pi_{PR} = (\{Z_0, Z_1, Z_2, Z_3, Z_4, N, Y, 0, 1, \Phi, C\}, \{\Phi\}, [1]_1, \mathcal{L}_0, \{r_1, \ldots, r_{12}\})$$

$$\mathcal{L}_0 = \{(Z_0, 2), (0, 1), (1, 1), (C, 1000)\}$$

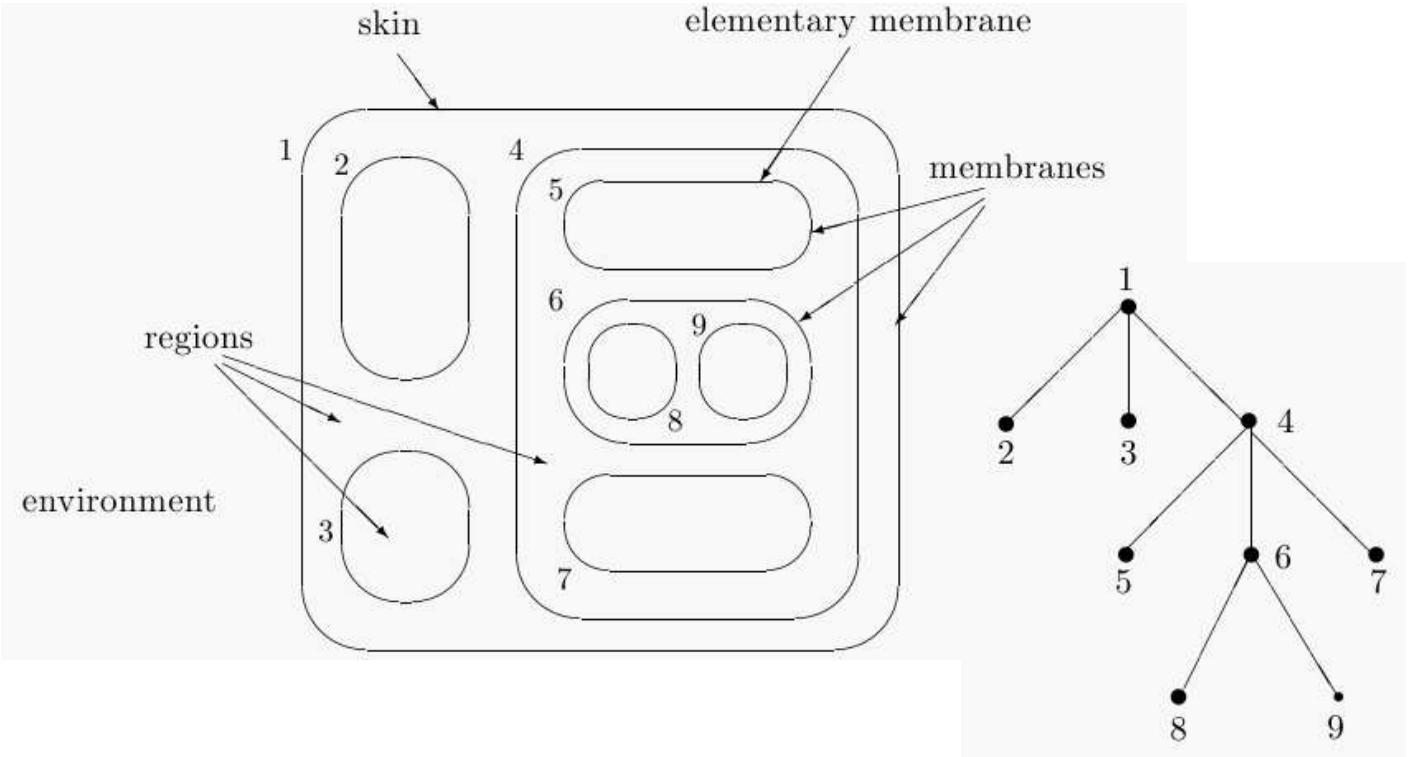
$r_1 : Z_3 + 1 + C \longrightarrow Y + \Phi + 1$
$r_2 : Z_4 + 0 + C \longrightarrow Y + \Phi + 0$
$r_3 : Y + 1 + C \longrightarrow Y + \Phi + 1$
$r_4 : Y + 0 + C \longrightarrow Y + \Phi + 0$
$r_5 : Z_0 + 1 + C \longrightarrow Z_1 + 1 + Z_0$
$r_6 : Z_0 + 0 + C \longrightarrow Z_2 + 0 + Z_0$
$r_7 : Z_1 + 1 + C \longrightarrow N + 1$
$r_8 : Z_1 + 0 + C \longrightarrow Z_4 + 0$
$r_9 : Z_2 + 1 + C \longrightarrow Z_3 + 1$
$r_{10} : Z_2 + 0 + C \longrightarrow N + 0$
$r_{11} : Z_3 + 0 + C \longrightarrow N + 0$
$r_{12} : Z_4 + 1 + C \longrightarrow N + 1$



Dynamical simulation was carried out using MatLab ($\Delta\tau = 10^{-4}$).
Example comes from transformation of a finite automaton.

From a Single Membrane to a Membrane Structure

Hierarchically nested membranes denoted as a tree



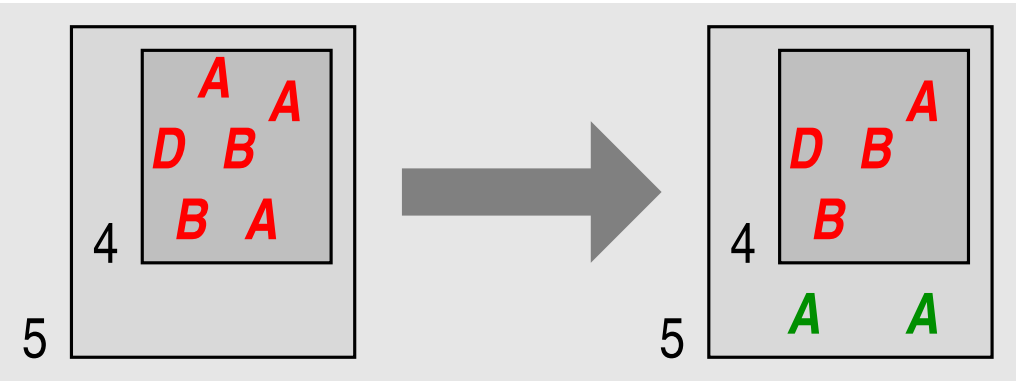
Representation as character string (or graph)

$$\mu = [_1[_2]_2[_3]_3[_4[_5]_5[_6[_8]_8[_9]_9]_6[_7]_7]_4]_1$$

G. Păun. Membrane Computing: An Introduction. Springer Verlag, 2002

Extensions of Rules Beyond Reactions: Transportation

Transportation of molecules through membranes



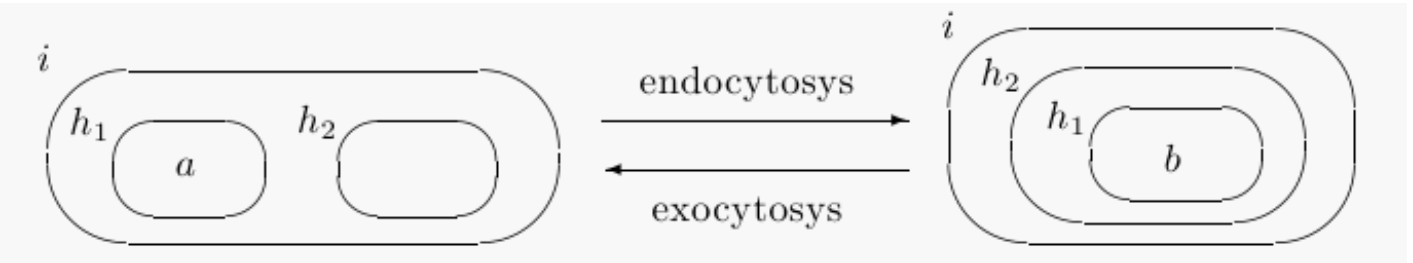
$$[_4]_4 : \{(A, 2)\} \longrightarrow [_5]_5$$

Algebraic settings available for different scenarios like

- Diffusion (unspecific transport through neighboured membranes)
- Receptors (acting as molecular filters)
- Symport/antiport (interactive molecular exchange)

Active Membranes

Rules for dynamical changes in membrane structure



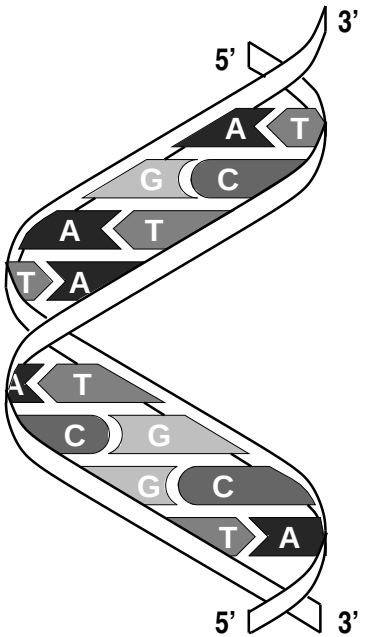
G. Păun. Introduction to Membrane Computing. Romanian Academy, 2001

$$[_{h_1} a]_{h_1} [_{h_2}]_{h_2} \longleftrightarrow [_{h_2} [_{h_1} b]_{h_1}]_{h_2}$$

Specification

- Change in membrane structure μ effects further system components
- Adjustment of configuration $\mathcal{L}$ and rules $\mathcal{R}$ attached to concerned membranes
- Membrane properties like electric charge (e.g. $[]^+, []^-$)

Structuring the Objects



DNA-Strang

linearisierte Darstellung

5' T C T A G A C G T 3'
3' A G A T C T G C A 5'

Erkennen der zugrunde liegenden DNA-Einzelstränge

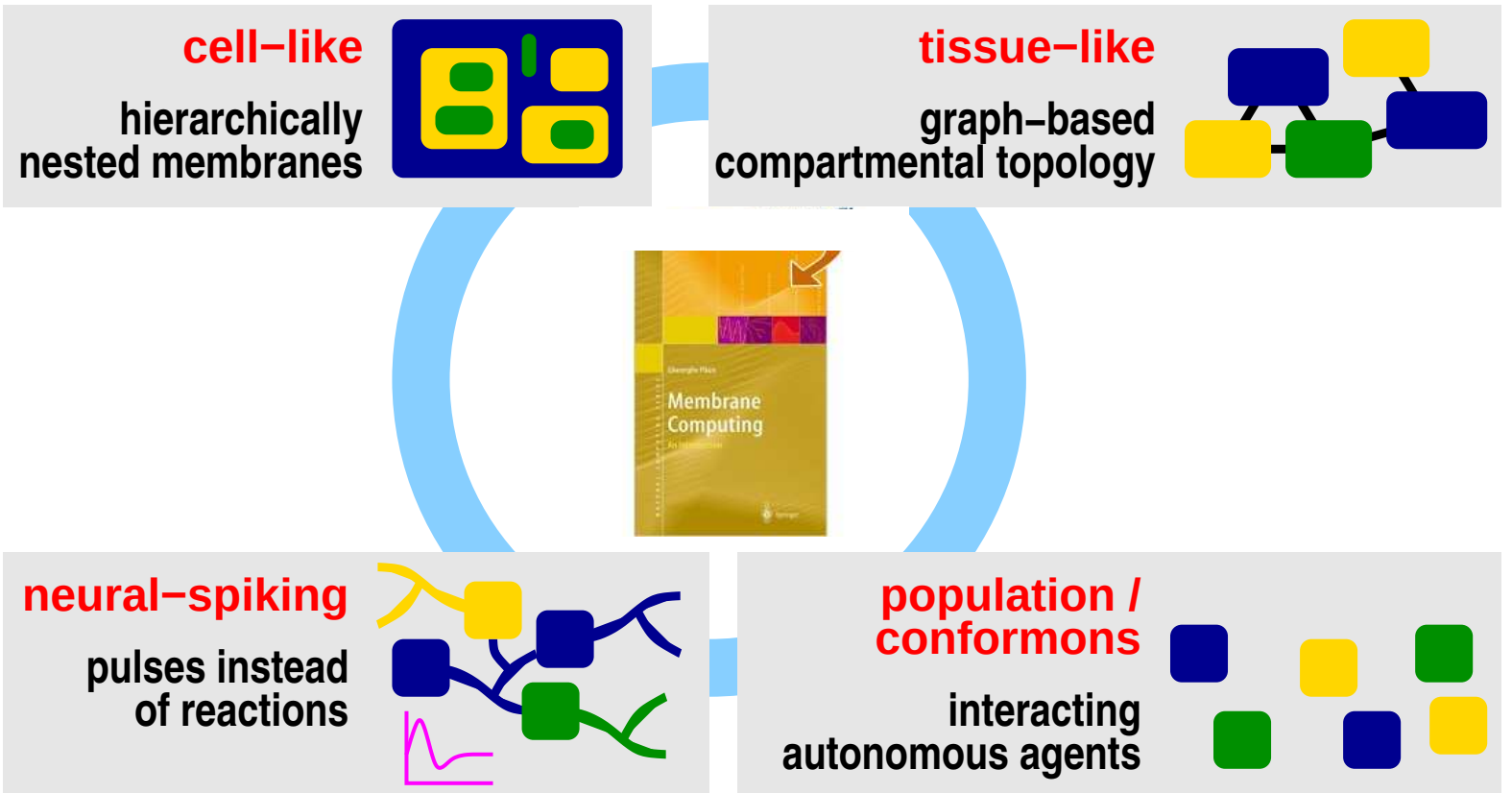
H-TCTAGACGT-H
H-ACGTCTAGA-H

Notation als Wörter einer formalen Sprache

T. Hinze, M. Sturm. Rechnen mit DNA. 316pp., Oldenbourg Wissenschaftsverlag, 2004

- Algebraic representation of molecular entities or reactants as **character strings** in terms of **formal languages**
- Placeholders ($*$) for handling **regular expressions** in conjunction with **matching strategies**

Classification of Membrane Systems

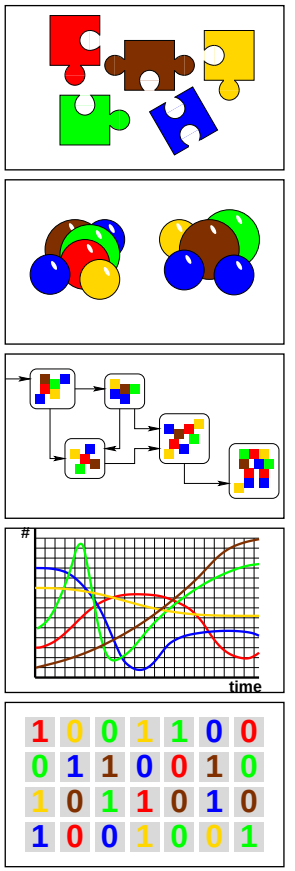


G. Păun. Membrane Computing: An Introduction. Springer Verlag, 2002

- Main classes with respect to compartmental topology and principle of operation

A Membrane System for Cell Signalling Processes

- Capture significant aspects of cellular signalling:
  - Components, topology, modularity
  - Protein activation states
  - Dynamical behaviour (kinetics)
  - Signal coding and transduction
  - Coping with incomplete protein information
- Based on tissue-like membrane systems
- Keep formalism tractable
- Balance detailedness with computational needs
- Facilitate system modification, recombination, and construction *ab initio*



T. Hinze, T. Lenser, P. Dittrich. A Protein Substructure Based P System for Description and Analysis of Cell Signalling Networks. In H.J. Hoogeboom, G. Paun, G. Rozenberg, A. Salomaa (Eds.), Membrane Computing. Series Lecture Notes in Computer Science 4361:409-423, Springer Verlag, 2006

System Definition Π_{CSN}

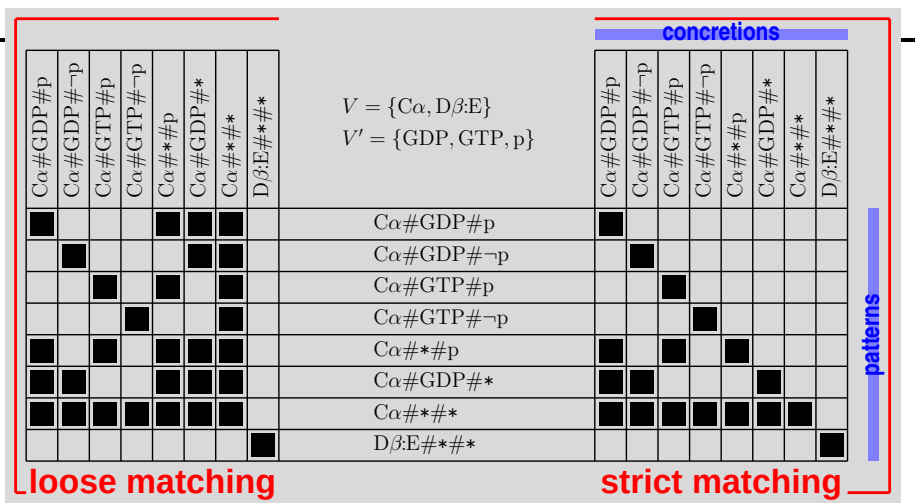
- System $\Pi_{CSN} = (V, V', E, M, n)$
  - V: alphabet of protein identifiers
  - V': alphabet of protein substructure/property identifiers
  - M: **modules** $\longrightarrow$ functional reaction units
  - E: graph $\longrightarrow$ transduction **channels** between modules
  - n: number of modules (degree of the P system)
- Modules $M_i = (R_{i1}, \ldots, R_{ir_i}, \mathrm{f}_{i1}, \ldots \mathrm{f}_{ir_i}, A_i) \in M$
  - R_{ij}: reaction rule $\longrightarrow$ multisets of educts and products may contain meta-symbols $\longrightarrow$ **matching** required
  - f_{ij}: function corresponding to **kinetics** of R_{ij}, number of educt objects taken from module within one reaction step
  - A_i: multiset of axioms $\longrightarrow$ initial contents of M_i
- Channels $e_{ij} = (i, j, I_{ij}, \mathrm{d}_{ij}) \in E$
  - Weighted directed channel from module i to module j
  - I_{ij}: filter interface (**receptor pattern** and **conc. gradient**)
  - d_{ij}: time **delay** (number of system **steps**) for passage

Matching and Matching Strategies

- String-based representation of proteins
  - **String-object** s: representation of a protein, its properties, substructure, binding domains, activation state, ligands
  - **Structure**: $s \in V^+ \otimes (\{\#\} \otimes ((V')^+ \cup \{\neg\} \otimes (V')^+ \cup \{*\}))^*$
  - **Meta-symbols**: **placeholder** (wild-card) $*$ $\longrightarrow$ appropriate or unknown substructure/property; **exclusion** $\neg$
  - Test whether two string-objects identify same molecule
- Matching strategies

loose:

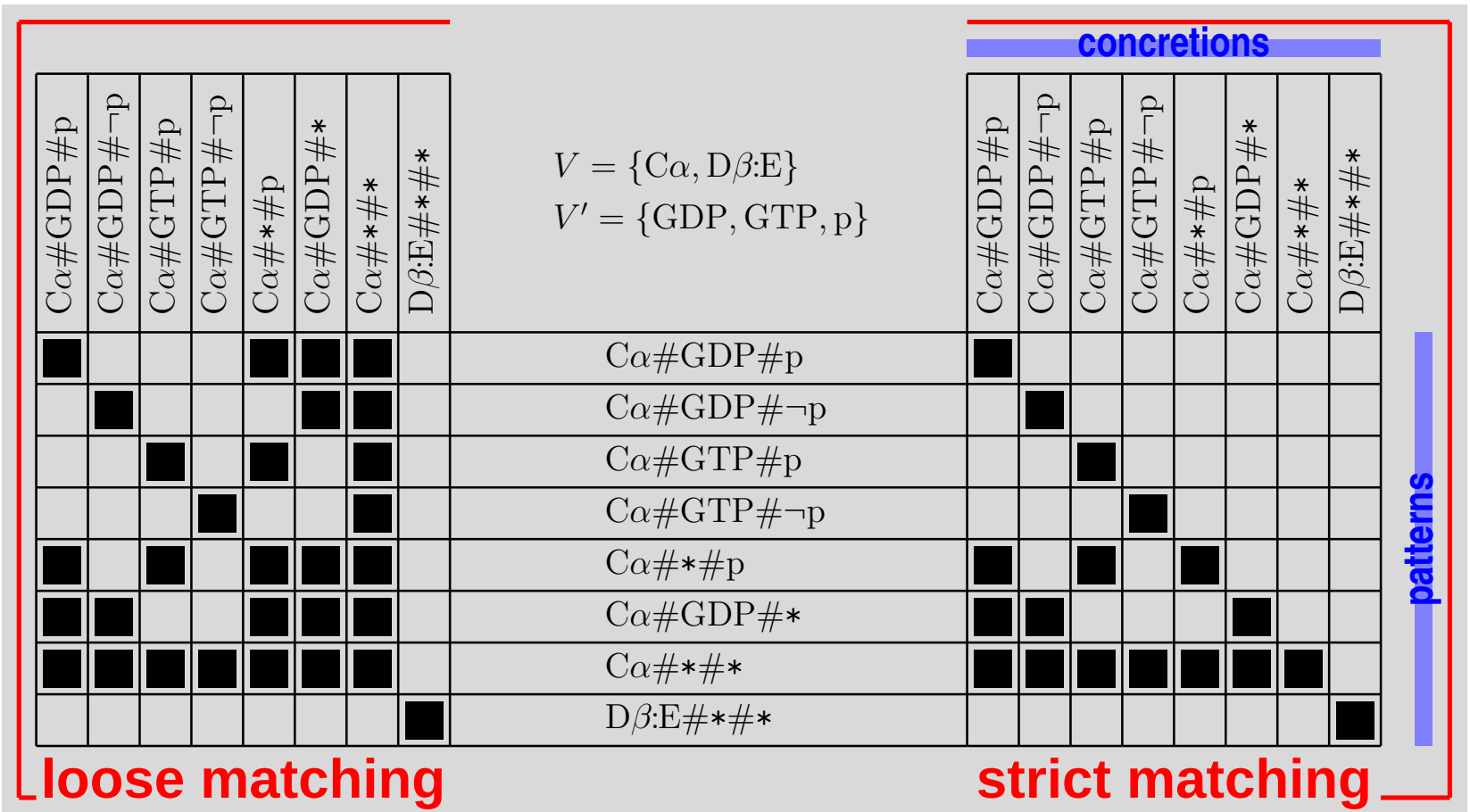
two string-objects match iff there is at least one common wild-card free representation



strict:

participating string-objects interpreted as a pattern and a candidate (concretion of the pattern)

Matching and Matching Strategies



System Behaviour and Properties

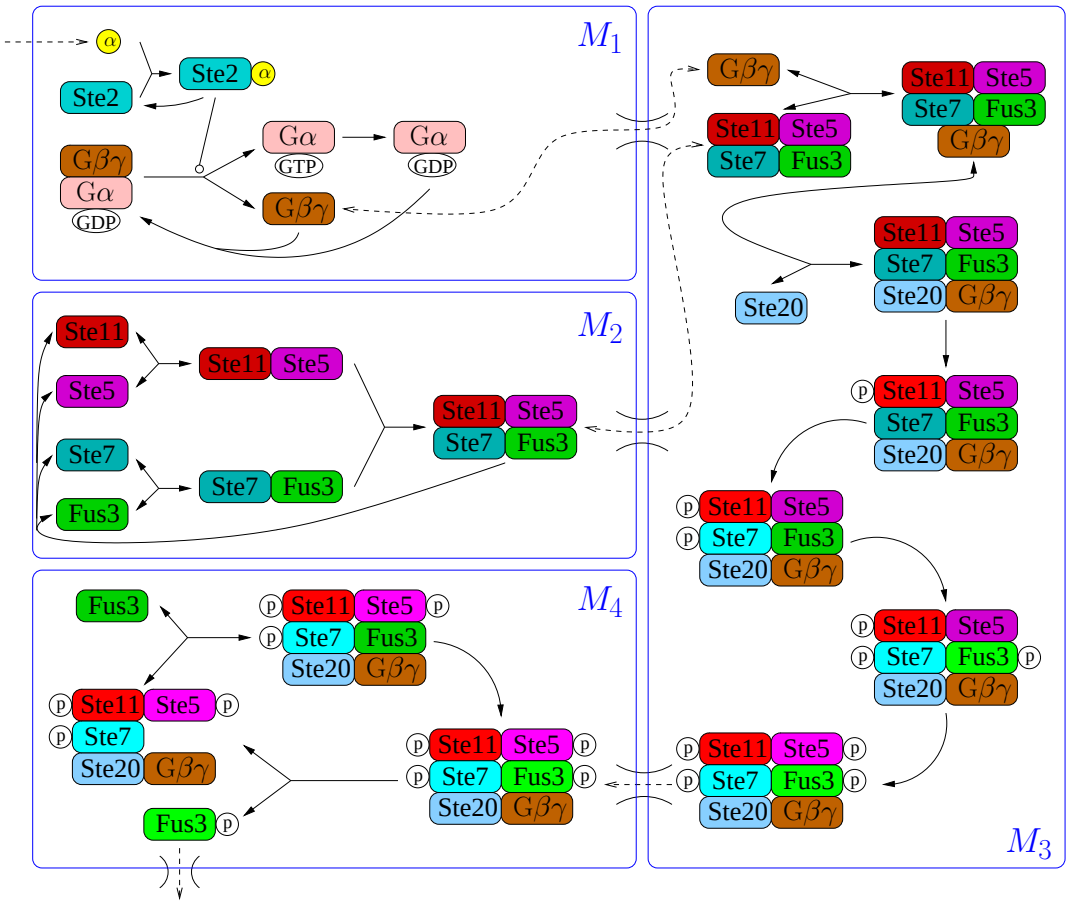
- Definition of dynamical system behaviour
  - Contents of module M_i at global time $t \in \mathbb{N}$: multiset $L_i(t)$
  - System step by module M_i:

$$\begin{aligned} L_i(0) &= A_i \\ L_i'(t) &= L_i(t) \ominus \mathit{Educts}_i(t) \uplus \mathit{Products}_i(t) \\ L_i(t+1) &= L_i'(t) \ominus \mathit{Outgoing}_i(t) \uplus \mathit{Incoming}_i(t) \end{aligned}$$

    1. Determine multiset of educts using $L_i(t)$, $R_{i1}, \ldots, R_{ir_i}, \mathrm{f}_{i1}, \ldots \mathrm{f}_{ir_i}$; involves matching
    2. Remove educt objects from module contents
    3. Determine and add multiset of reaction products, obtain $L_i'(t)$
    4. Determine and separate objects leaving host module evaluate $L_i'(t)$ and I for each outgoing channel, matching
    5. Add objects received from incoming channels, consider d
- System properties
  - Modularity – static system topology – ability to identify objects/substructures – flexibility in level of abstraction
  - Determinism – computational tractability – universality

Example: Yeast Pheromone Pathway

Signal transduction in *Saccharomyces cerevisiae*



$$\Pi = (V, V', E, M, 4)$$

$$V = \{\mathrm{Ste2}, \alpha, \mathrm{G}\beta\gamma, \mathrm{G}\alpha, \ldots\}$$

$$V' = \{\mathrm{a}, \mathrm{GDP}, \mathrm{GTP}, \mathrm{p}\}$$

$$M = \{M_1, M_2, M_3, M_4\}$$

$$M_1 = (R_{11}, R_{12}, R_{13}, R_{14}, R_{15}, \mathrm{f}_{11}, \mathrm{f}_{12}, \mathrm{f}_{13}, \mathrm{f}_{14}, \mathrm{f}_{15}, A_1)$$

$$R_{11} = \mathrm{Ste2}\#\neg\mathrm{a} + \alpha \rightarrow \mathrm{Ste2}\#\mathrm{a}$$

$$R_{12} = \mathrm{Ste2}\#\mathrm{a} \rightarrow \mathrm{Ste2}\#\neg\mathrm{a}$$

$$\mathrm{f}_{11} = \lfloor \mathrm{k}_{11}[\mathrm{Ste2}\#\neg\mathrm{a}][\alpha]/\mathrm{V}_1^2 \rfloor$$

$$\vdots$$

T. Hinze, T. Lenser, P. Dittrich. A Protein Substructure Based P System for Description and Analysis of Cell Signalling Networks. In H.J. Hoogeboom, G. Paun, G. Rozenberg, A. Salomaa (Eds.), Membrane Computing. Series Lecture Notes in Computer Science 4361:409-423, Springer Verlag, 2006

Bio-Applications: Further Appetizers

Bio-applications

2. P System Models for Mechanosensitive Channels
Ioan I. Ardelean, Daniela Besozzi, Max H. Garzon, Giancarlo Mauri, Sujoy Roy 43

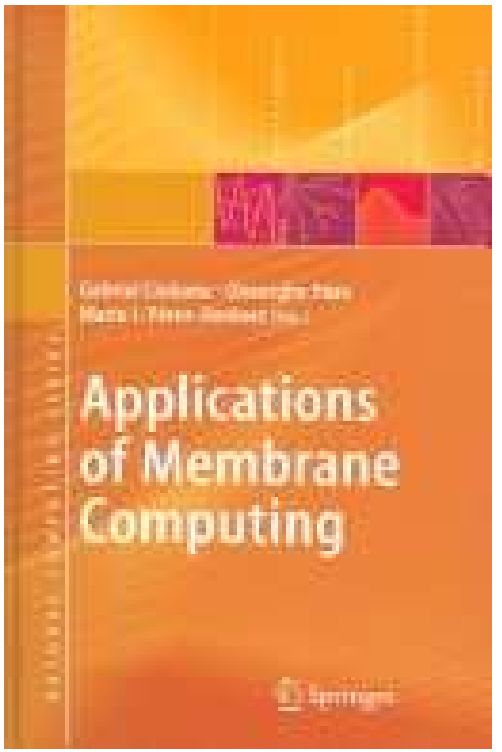
3. P Systems for Biological Dynamics
Luca Bianco, Federico Fontana, Giuditta Franco, Vincenzo Manca 83

4. Modeling Respiration in Bacteria and Respiration/Photosynthesis Interaction in Cyanobacteria Using a P System Simulator
Matteo Cavaliere, Ioan I. Ardelean 129

5. Modeling Cell-Mediated Immunity by Means of P Systems
Gabriel Ciobanu 159

6. A Membrane Computing Model of Photosynthesis
Taishin Yasunobu Nishida 181

7. Modeling p53 Signaling Pathways by Using Multiset Processing
Yasuhiro Suzuki, Hiroshi Tanaka 203



books.google.de

G. Ciobanu, G. Păun, M.J. Perez-Jimenez (Eds.). Applications of Membrane Computing. Springer Verlag, 2005

Membrane Systems as Universal Computing Devices

Models and concepts for biologically inspired computing

- **(Bio)molecular computation**
- **Genetic circuits**
- **Cell-based computing**
- **Neural networks**
- **Gene assembly**
- **Evolutionary computing**
- **Amorphous computing**

Alan Turing
www.computerhistory.org

$\Longrightarrow$ Covered by variants of P systems

$\Longrightarrow$ Computational completeness shown by simulation of Turing machines or equivalents

Random Access Machine (RAM)

Established Turing-Complete Model for Computation

- Syntactical denotation of components
  $RAM = (R, L, P, l_1)$
  - l_1: jump label of first instruction
  - P: program (finite set of instructions)
  - L: finite set of jump labels $L = \{l_1, \ldots, l_c\}$
  - R: finite set of registers $R = \{r_1, \ldots, r_m\}, \quad r_k \in \mathbb{Z}$
- Available instructions
  - $l_i : \texttt{INCR}(r_k), l_j$ increment register r_k, jump to l_j
  - $l_i : \texttt{DECR}(r_k), l_j$ decrement register r_k, jump to l_j
  - $l_i : r_k > 0, l_j, l_p$ if $r_k > 0$ jump to l_j else jump to l_p
  - $l_i : \texttt{HALT}$ terminate program and output
- Useful assumptions
  - Consecutive indexing of jump labels and registers
  - Determinism
  - Initialisation of registers at start
  - Output of all m registers when HALT

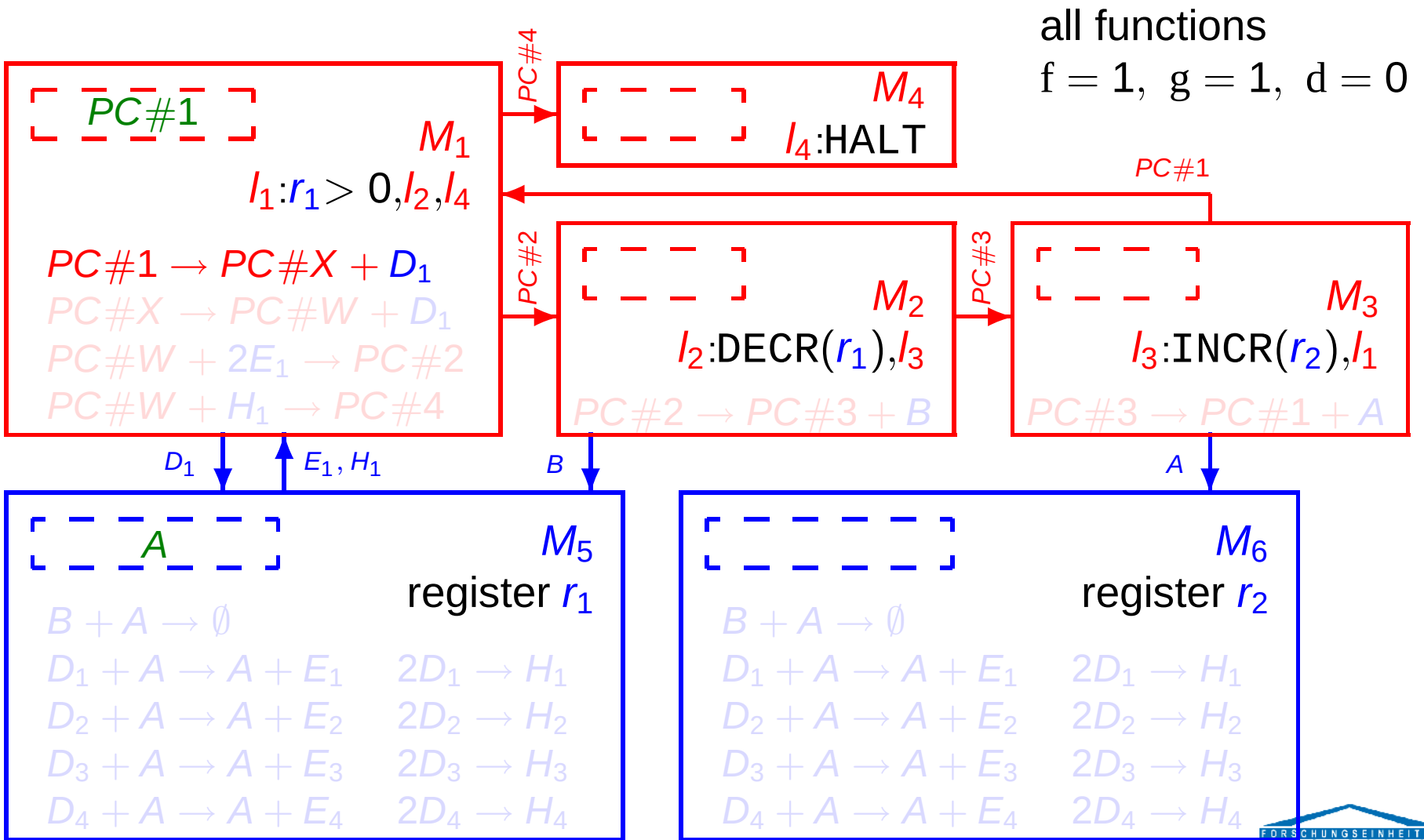
Simulation of Register Machines by Π_{CSN}

- Consider for simulation and transformation

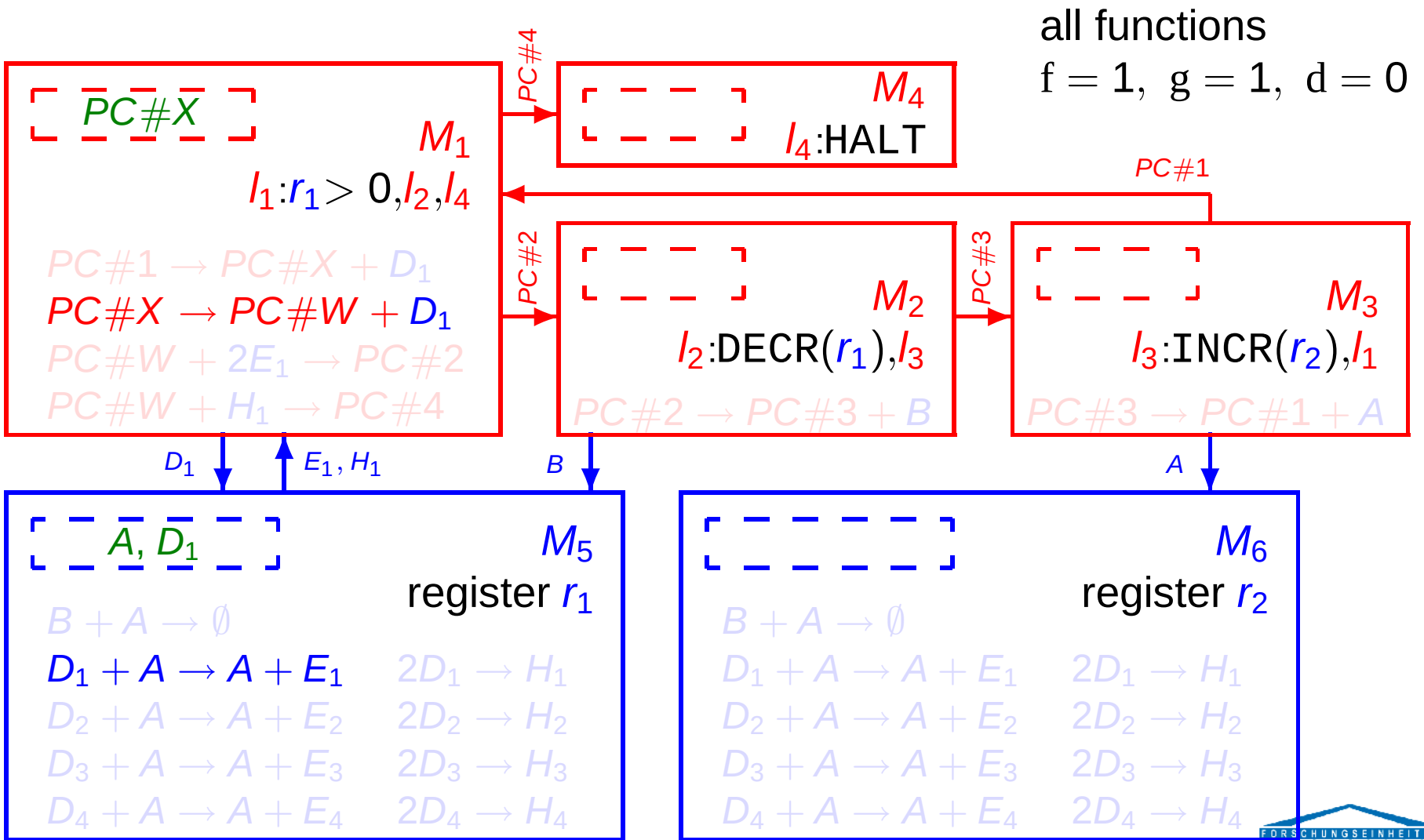
$$
\begin{aligned}
RAM &= (R, L, P, l_1) \\
R &= \{r_1, r_2\} \\
L &= \{l_1, l_2, l_3, l_4\} \\
P &= \{l_1 : r_1 > 0, l_2, l_4, \\
&\quad\ l_2 : \texttt{DECR}(r_1), l_3, \\
&\quad\ l_3 : \texttt{INCR}(r_2), l_1, \\
&\quad\ l_4 : \texttt{HALT}\}
\end{aligned}
$$

- Initialisation: $r_1 = 1$ and $r_2 = 0$ (arbitrarily chosen)
- Program moves contents of r_1 cumulatively to r_2
- RAM consists of $m = 2$ registers and $c = 4$ instructions
- Each register and each instruction forms separate module of membrane system $\Pi_{\text{CSN}} = (V, V', E, M, c + m)$

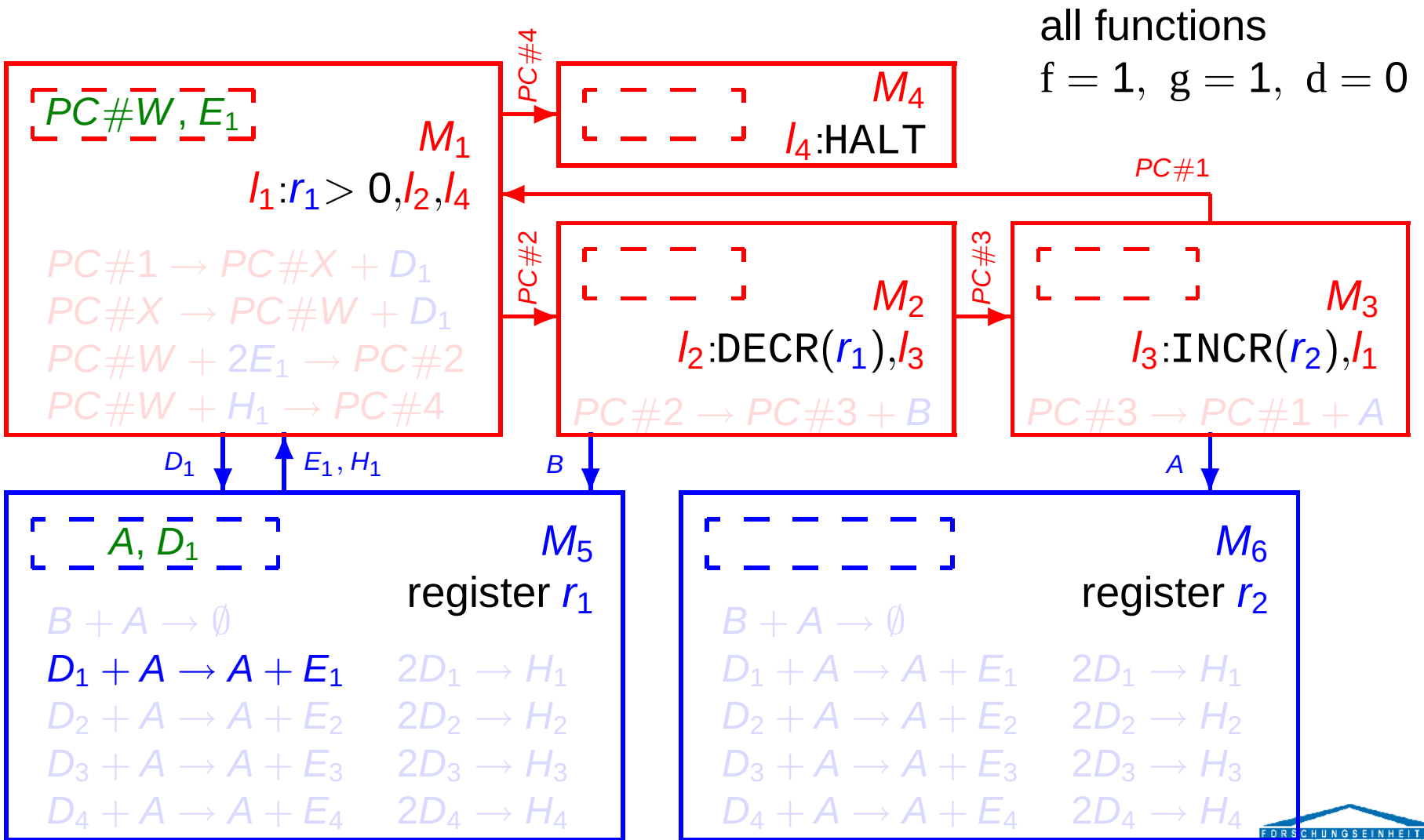
Simulation of Register Machines by Π_{CSN}



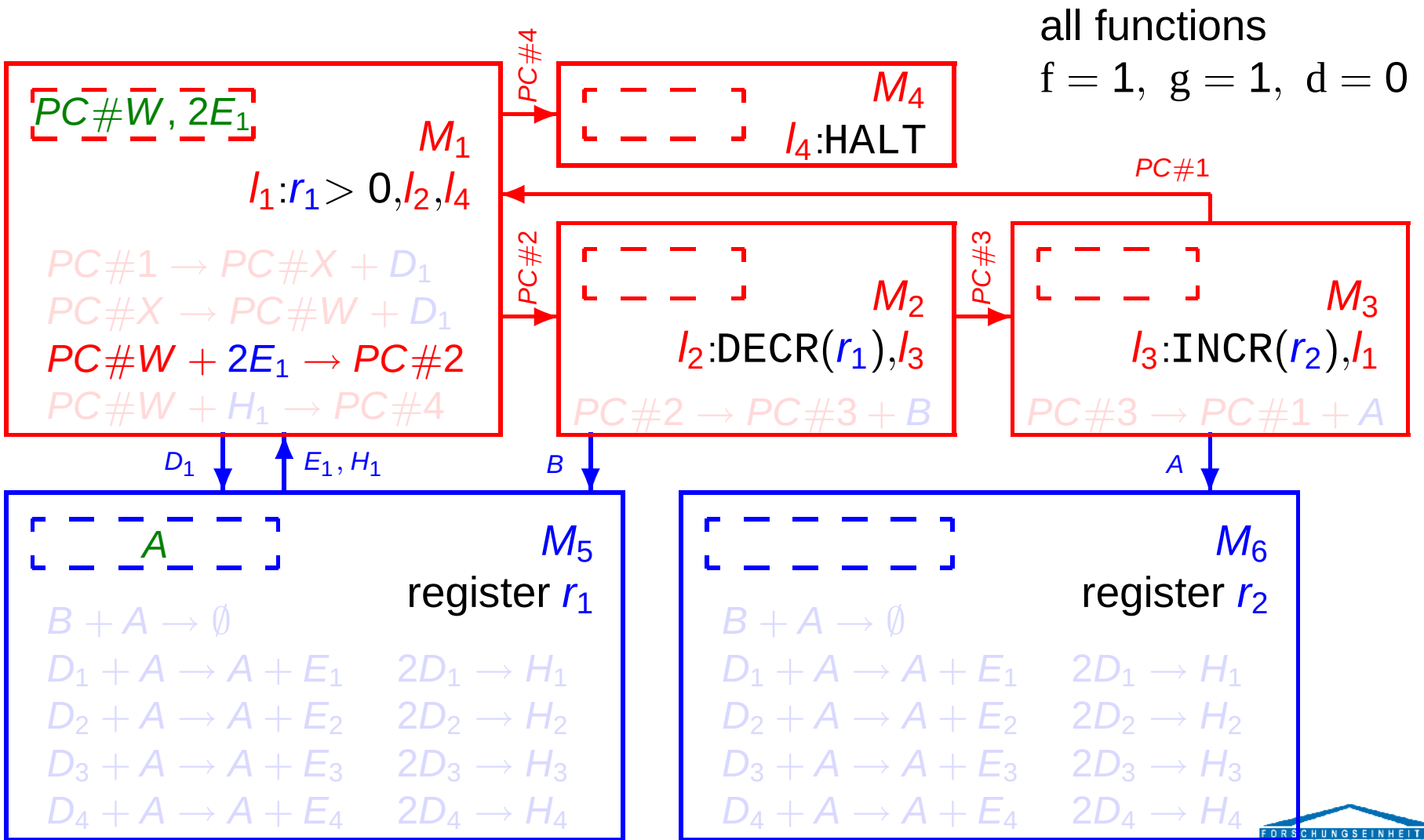
Simulation of Register Machines by Π_{CSN}



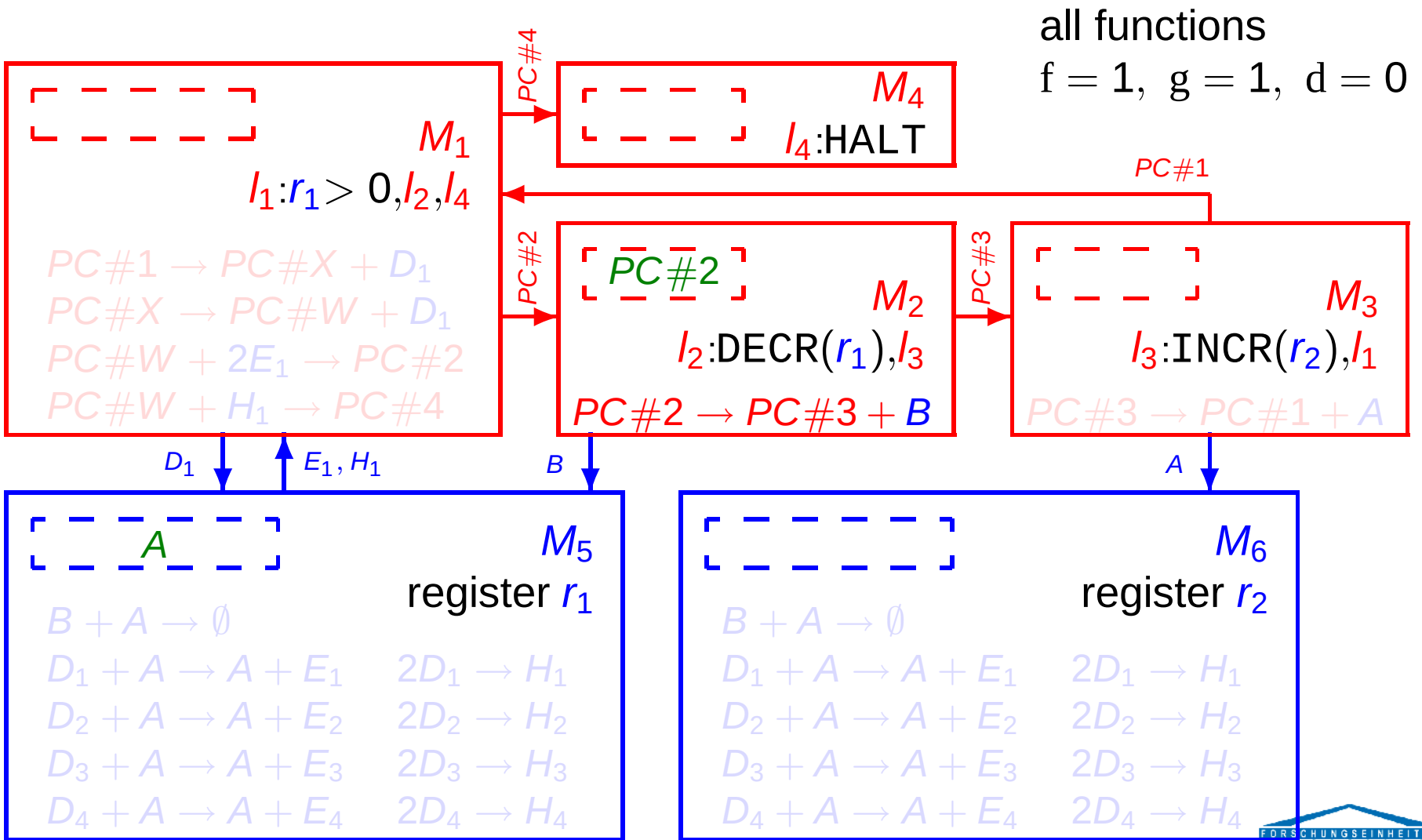
Simulation of Register Machines by Π_{CSN}



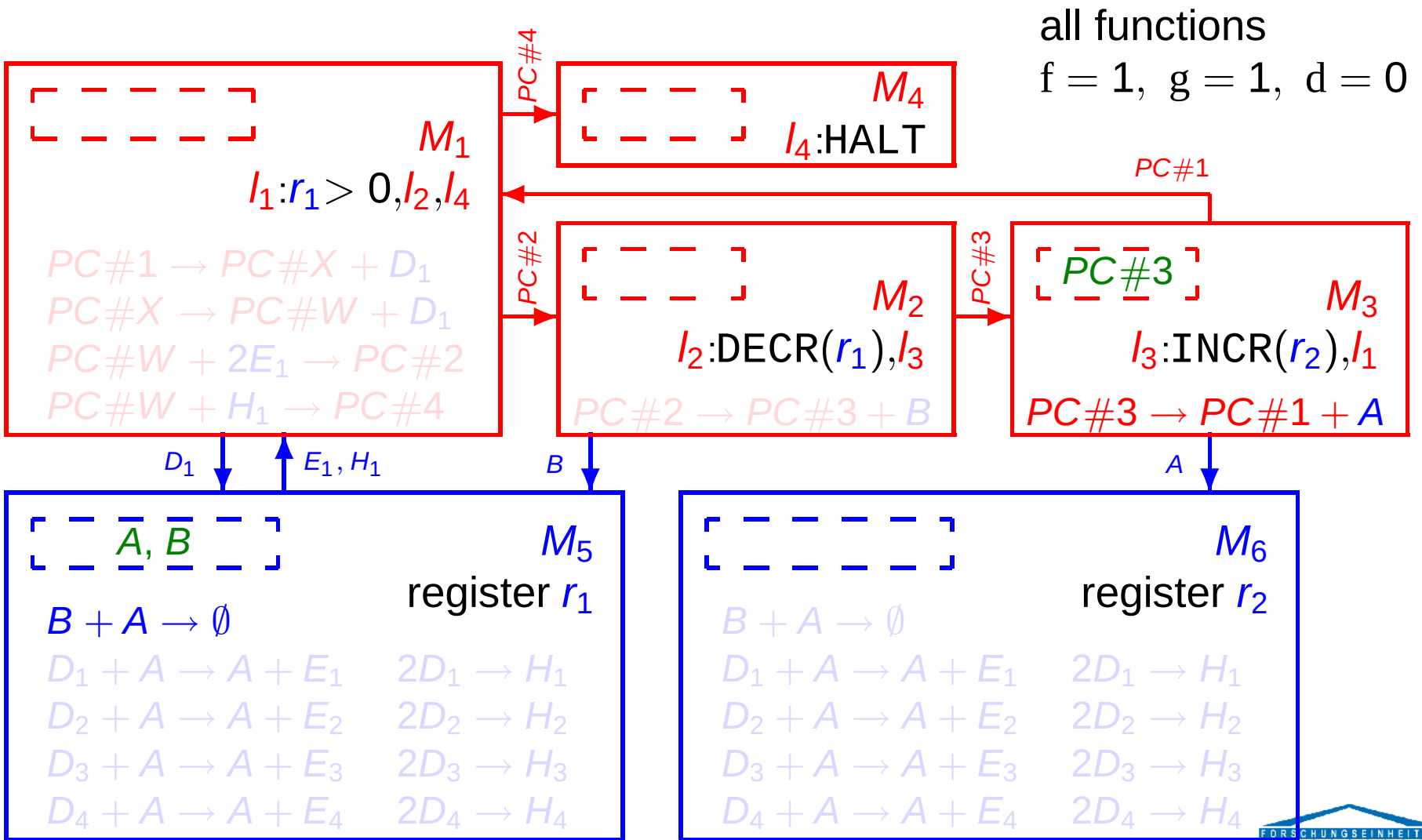
Simulation of Register Machines by Π_{CSN}



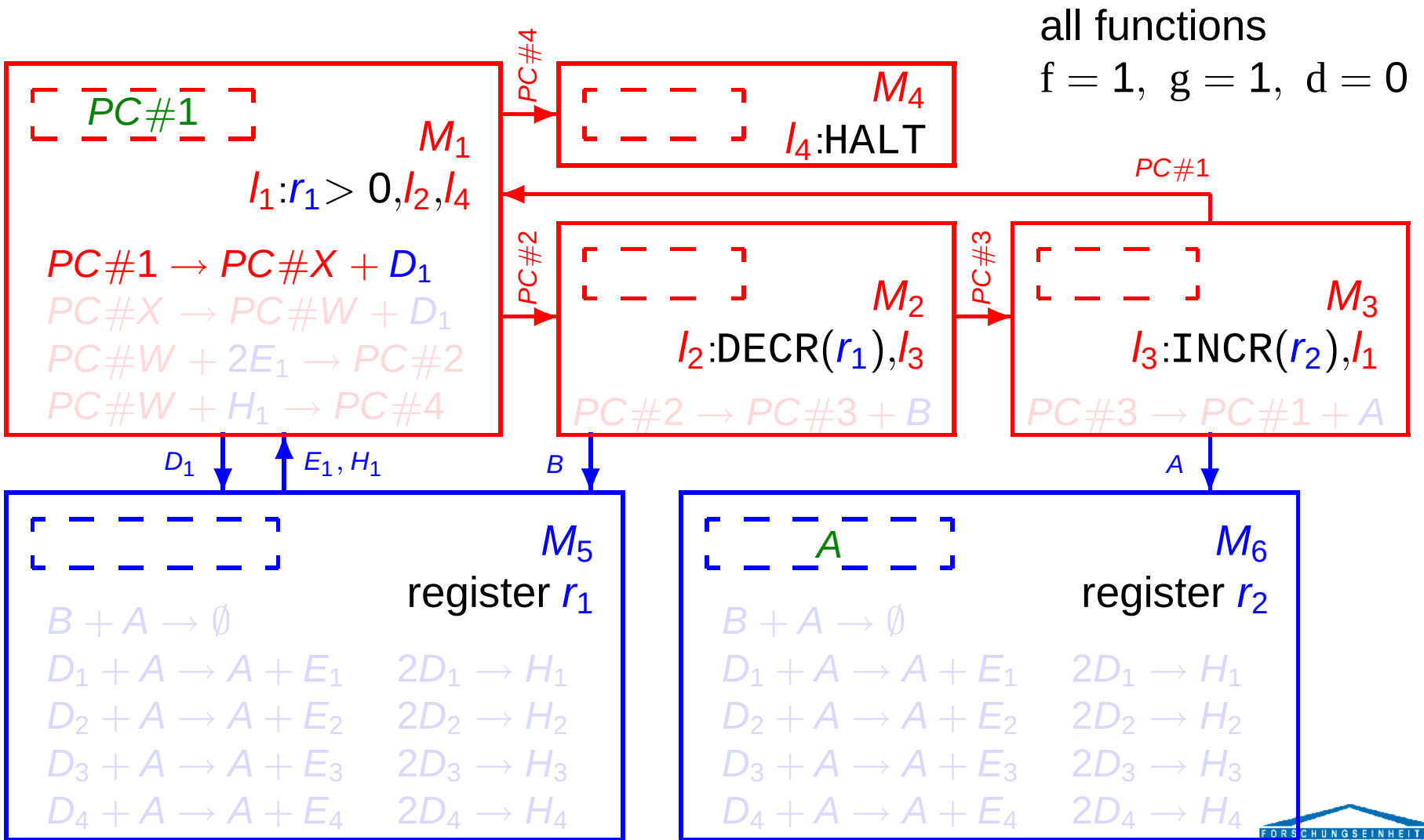
Simulation of Register Machines by Π_{CSN}



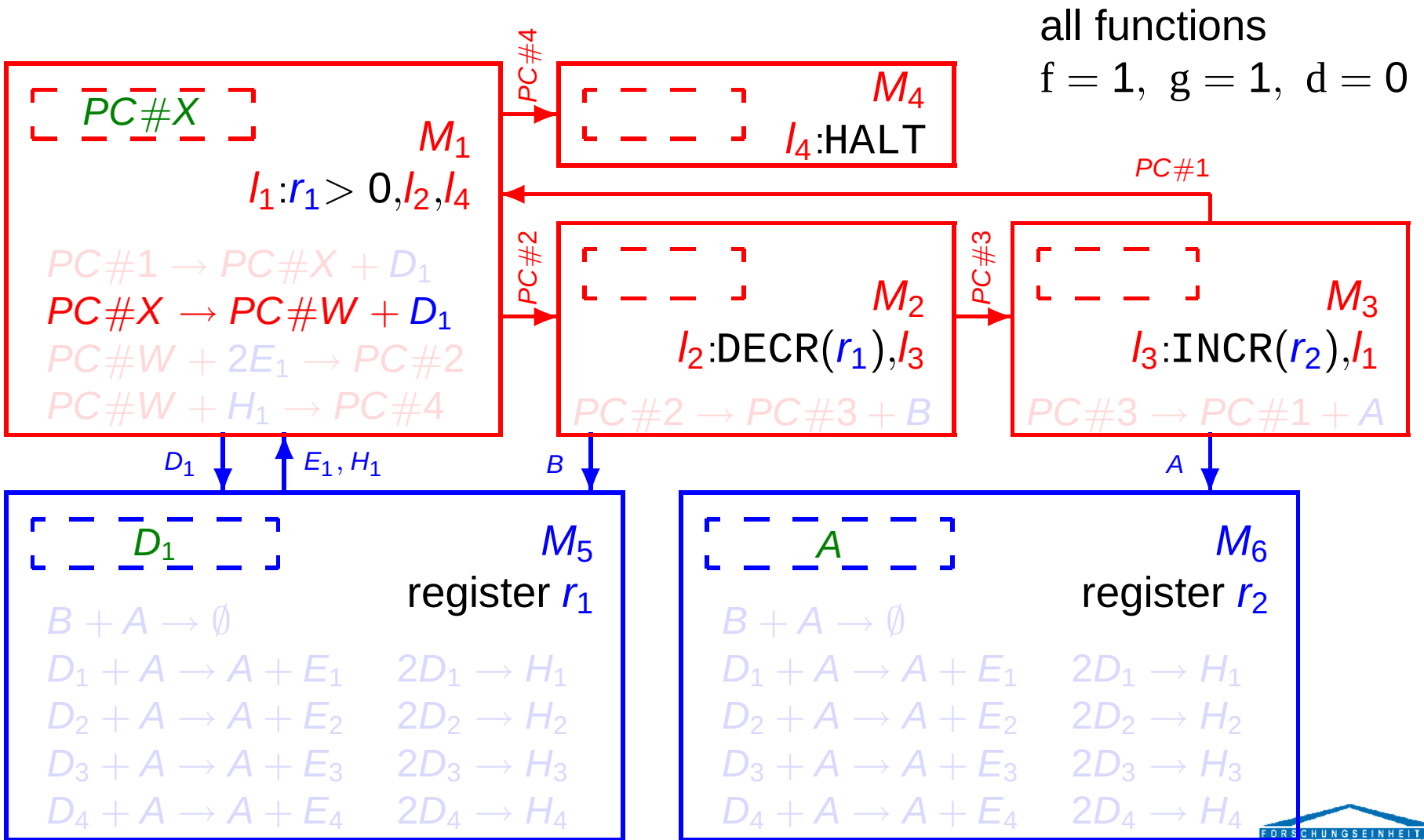
Simulation of Register Machines by Π_{CSN}



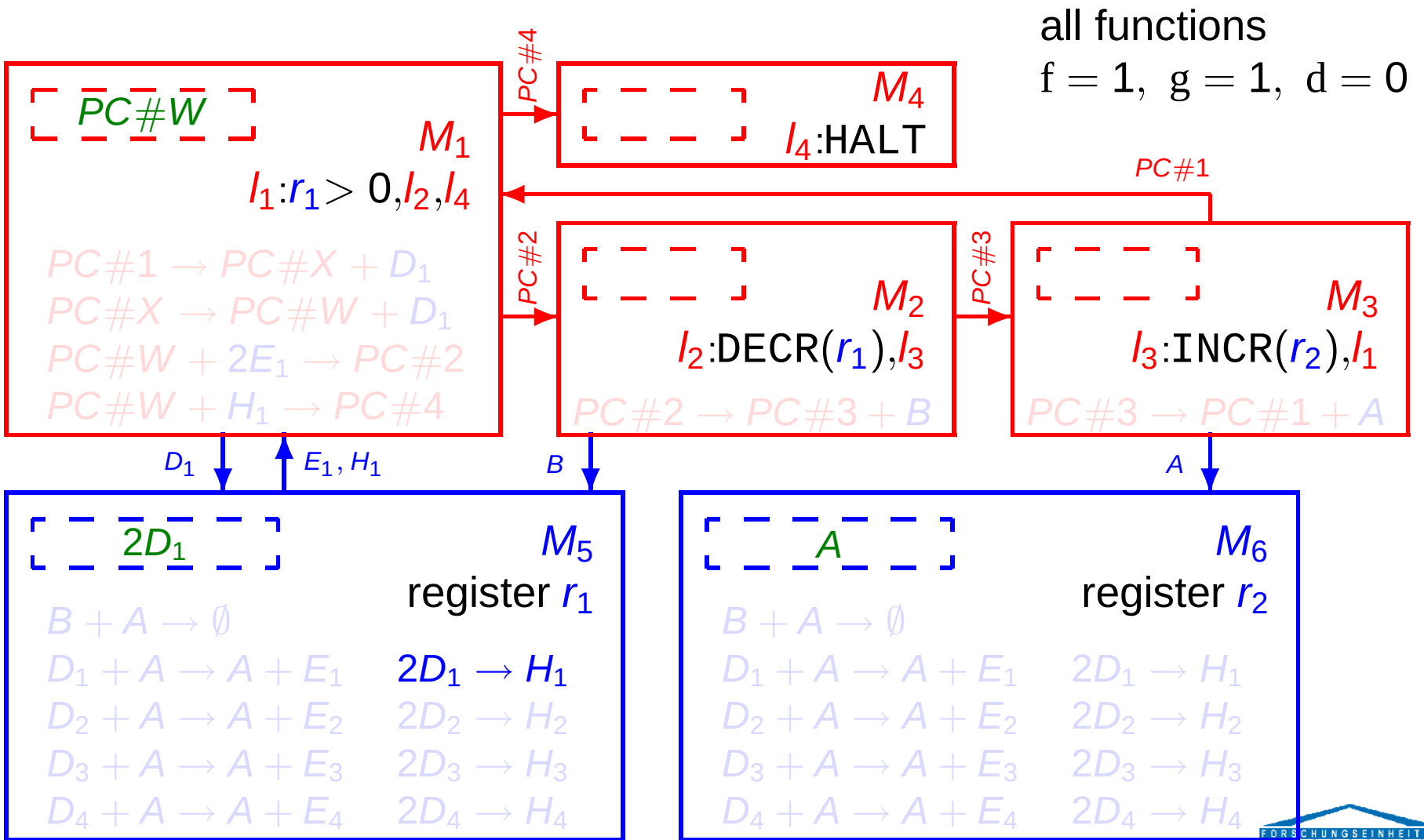
Simulation of Register Machines by Π_{CSN}



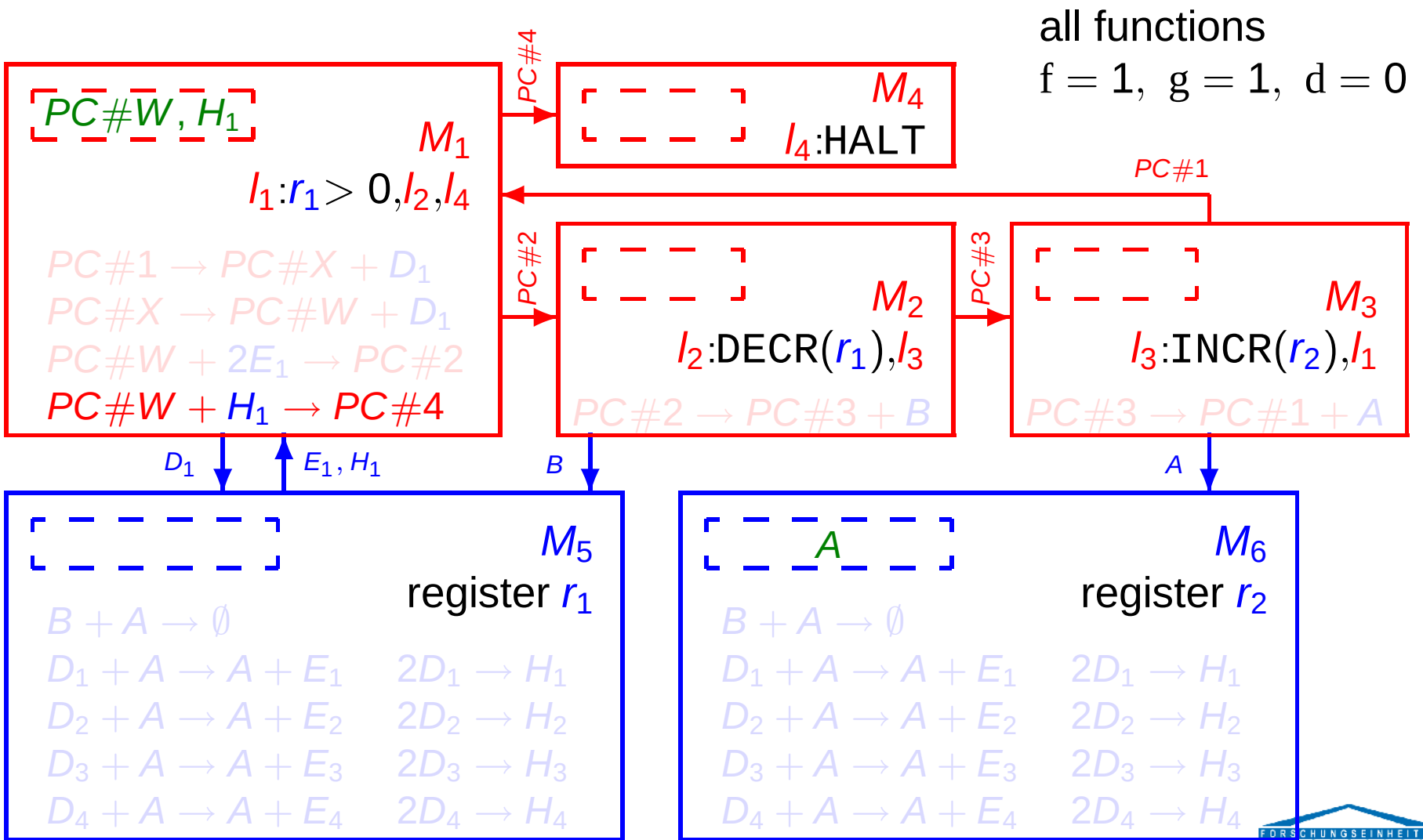
Simulation of Register Machines by Π_{CSN}



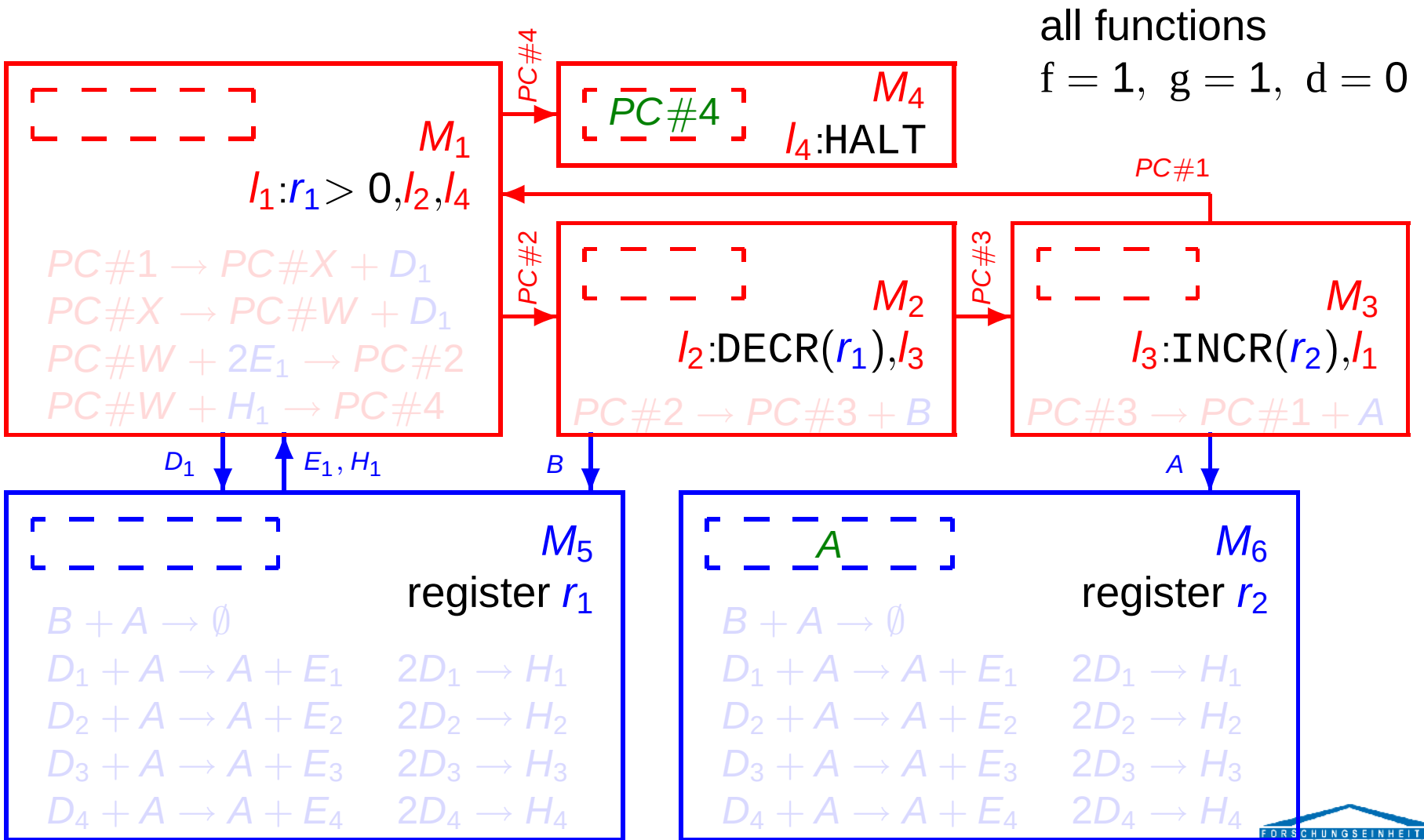
Simulation of Register Machines by Π_{CSN}



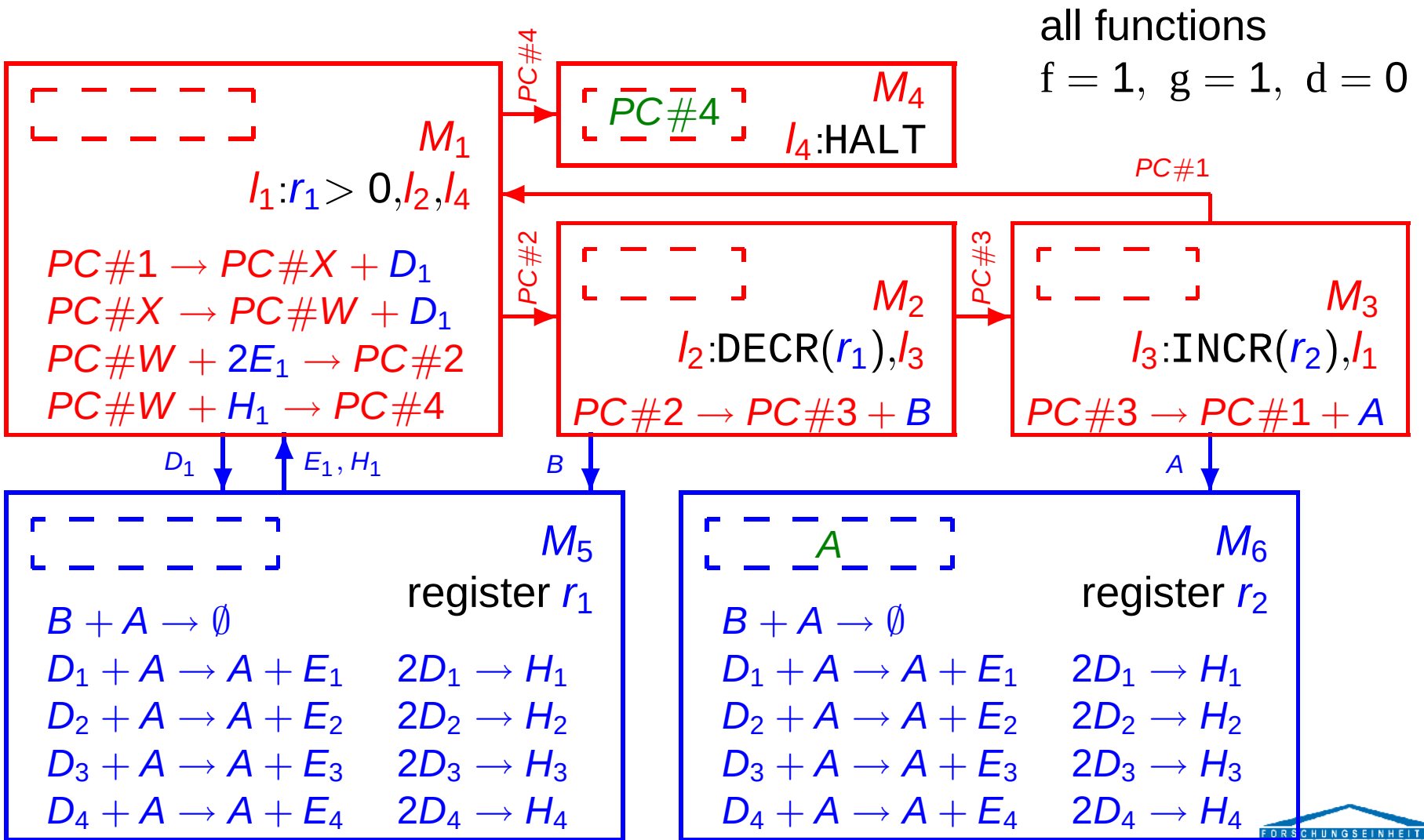
Simulation of Register Machines by Π_{CSN}



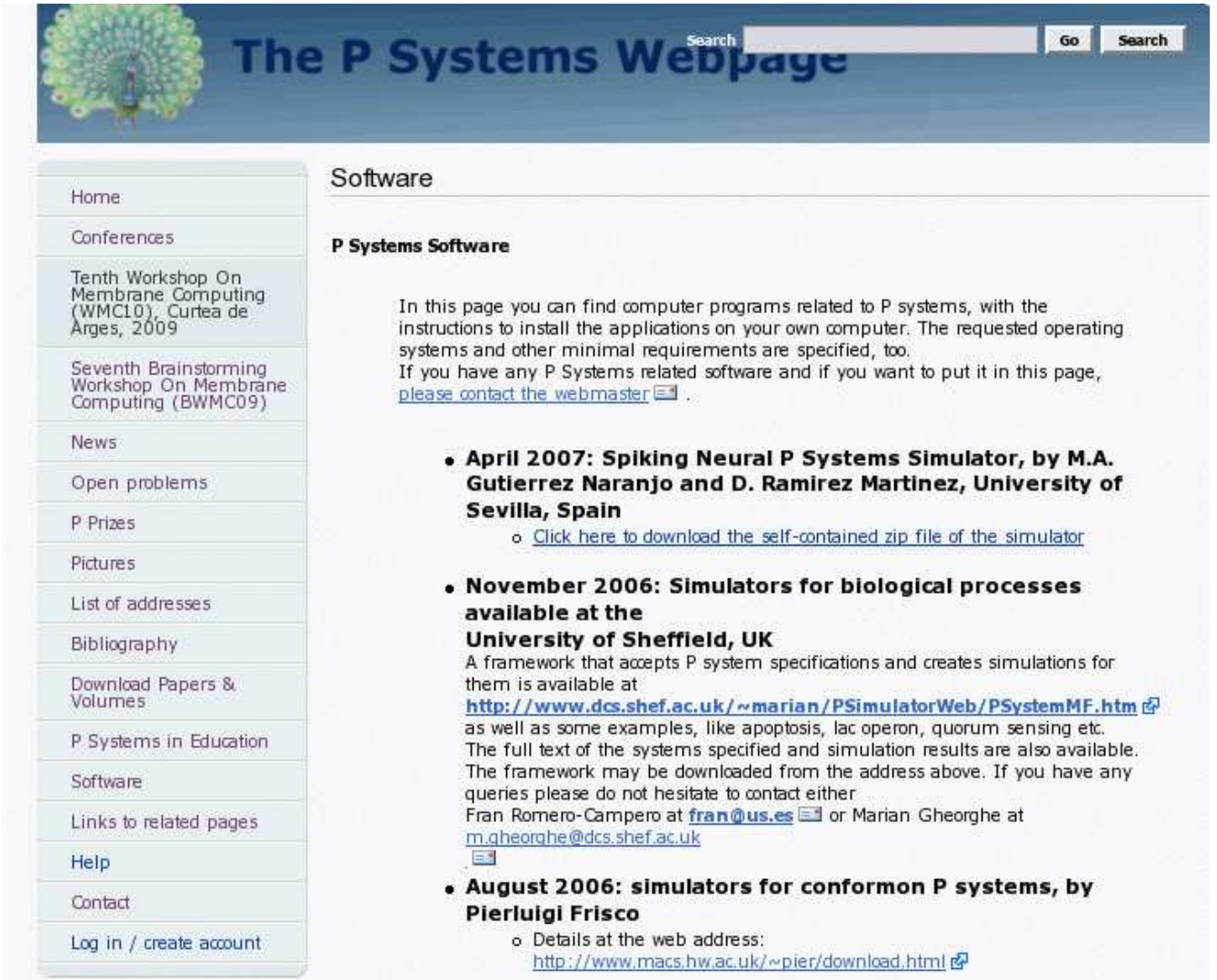
Simulation of Register Machines by Π_{CSN}



Simulation of Register Machines by Π_{CSN}



The P Page: A Repository and More . . .

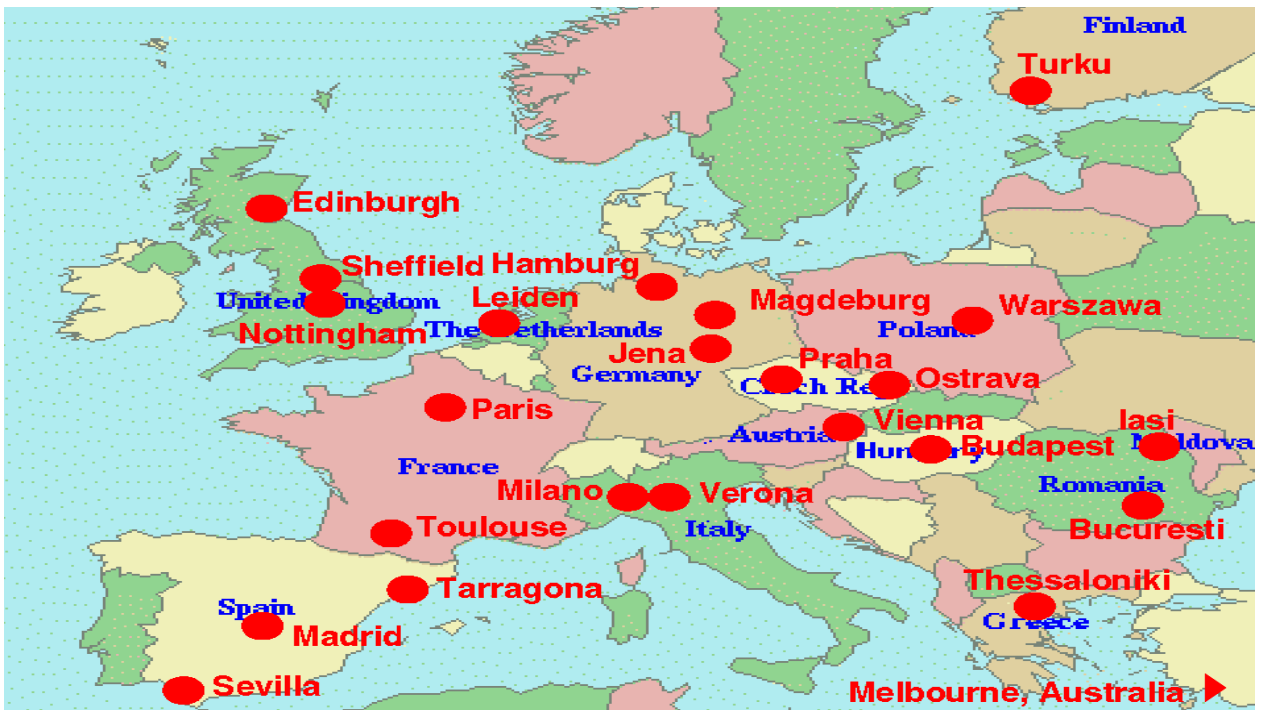


http://ppage.psystems.eu

Membrane Systems: International Dimension

- Pioneered in 1998 by Gheorghe Paun
- $\approx$ 10 years scientific evolution
- $\approx$ 100 active researchers in community
- $\approx 1,500$ publications up to now

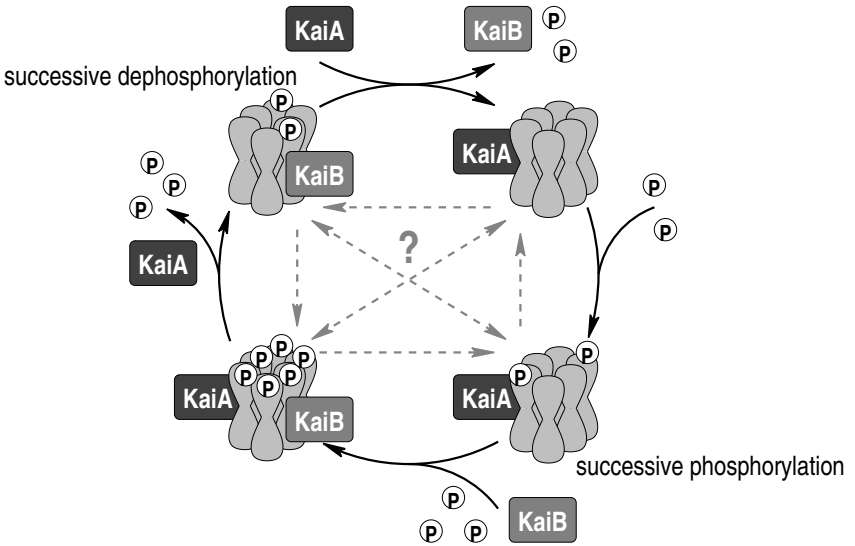
Gheorghe Păun
www.imar.ro/∼paun



Quo Vadis Membrane Systems? Trends and Visions

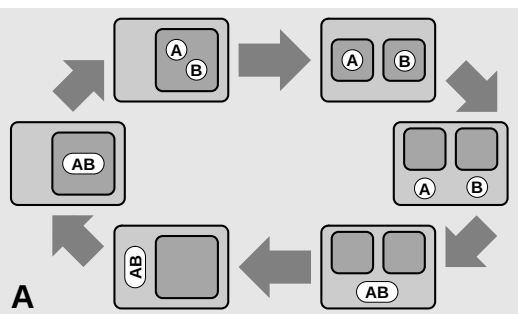
State-of-the-art

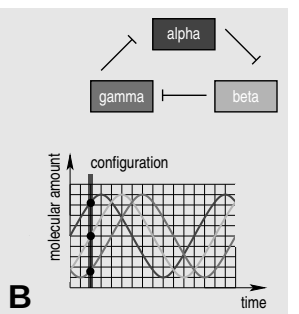
- Establish modelling technique in Systems Biology
- Capturing aspects of inter/ intracellular information proc.
- Approximation towards further bio-applications



Upcoming contributions

- Framework for **reverse engineering** (e.g. artificial evolution)
- **Backtracking** P systems
- **Molecular tracing**





Figures are parts of recently submitted material.

Envisioned: “Membrane Theory"