Master 2 MPRI course 2-19 Biochemical Programming
Jérôme Feret

Written examination 3 hours
All printed documents allowed
Any electronic devide prohibited
March 8th 2022

Abstract

This problem investigates the impact of structural symmetries on the stochastic behavior of Kappa models. Structural symmetries emerge from equivalence relationships among pairs of sites. That is to say sites having exactly the same capabilities of interaction. We will consider two case studies. In the first one, we introduce a model in which two sites have the same capabilities of interaction in any context. In the second one, we examine a model in which two sites have the same capabilities of interaction only when a third one is activated. We study the impact of these symmetries under the lens of forward bisimulations (which enable to quotient the underlying transition system by discarding the difference between symmetric states) and backward bisimulations (which highlight statistical invariants).

1 Weighted transition systems

Firstly, we introduce the notion of weighted transition systems for describing Markov chains. To make the things simpler, we consider finite Markov chains with discrete time evolution (finite DTMC) only.

Definition 1.1 *A weighted transition system is a pair $(\mathcal{Q}, w)$ where $\mathcal{Q}$ is a finite set of elements, called states, and w is a function mapping every pair of states to a non negative real numbers (in $\mathbb{R}_{\geqslant 0}$) such that for every state $q \in \mathcal{Q}$, the sum $\sum_{q' \in \mathcal{Q}} w(q, q')$ is equal to* 1.

In fact, for every state q, the function mapping every state q' to the real value $w(q, q')$ is the finite probability distribution for the next state of the system. Whenever $w(q, q') = 0$ we say that there is no transition from the state q to the state q', otherwise we say that there is a transition from the state q to the state q' with probability $w(q, q')$. An example of weighted transition system is depicted in Fig. **??**. States are described as ellipses labeled by names whereas transitions are denoted as edges labelled with their probabilities.

We assume until the rest of the section that we are given $(\mathcal{Q}, w)$ a weighted transition system.

Definition 1.2 (Trace) *A (finite) trace is a finite sequence of elements of the set $\mathcal{Q}$.*
The length of a trace is the number of states minus 1.
The probability of the trace $\tau \triangleq (q_i)_{0 \leqslant i \leqslant n}$ is defined as follows:

$$P(\tau \mid q_0) \triangleq \prod_{1 \leqslant i \leqslant n} w(q_{i-1}, q_i).$$

As a direct consequence, a trace has probability 0 whenever it contains two consecutive states not related by any transition. Moreover, we notice that traces of length 0 have probability 1.

Now we define the notion of flow between two sets of states.

Definition 1.3 (Flow) *The flow $\textsc{flow}(X, X')$ from a set of states $X \subseteq \mathcal{Q}$ into a set of states $X' \subseteq \mathcal{Q}$ is defined as follows:*

$$\textsc{flow}(X, X') \triangleq \sum_{q \in X, q' \in X'} w(q, q').$$

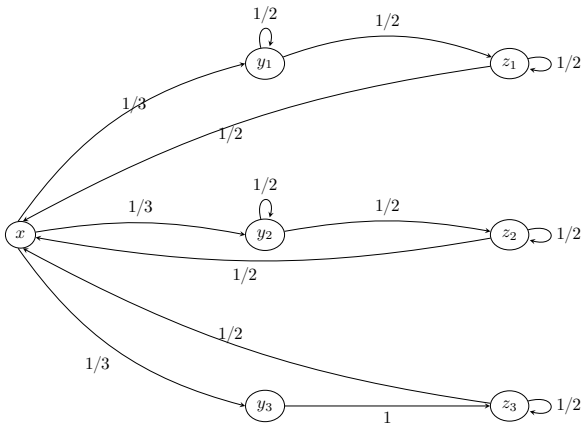


Figure 1: A weighted transition system.

2 Bisimulation over weighted transition system

2.1 Reminder on equivalence relations

The goal of this section is to introduce several notions of equivalence between the states of a weighted transition system. Our goal is to lump the states of the system accordingly.

Definition 2.1 (relation) *A (binary) relation over a set X is a subset of X^2.*

Whenever $\mathcal{R}$ is a relation over a set X, the notation $q \mathcal{R} q'$ stands for $(q, q') \in \mathcal{R}$.

Definition 2.2 (equivalence relation) *A relation $\mathcal{R}$ over a set X is an equivalence relation whenever it is reflexive, symmetric, and transitive.*
That is to say that, for every $x, y, z \in X$:

- *(reflexivity) $x \mathcal{R} x$;*
- *(symmetry) $x \mathcal{R} y \implies y \mathcal{R} x$;*
- *(transitivity) $[x \mathcal{R} y \wedge y \mathcal{R} z \Longrightarrow x \mathcal{R} z]$.*

Definition 2.3 *An equivalence relation is usually denoted as $\sim$. Given an equivalence relation $\sim$ over a set X and an element $x \in X$, the set of elements x' such that $x \sim x'$ is called the $\sim$-equivalence class of the element x and is denoted as $[x]_\sim$. The set of $\sim$-equivalence classes is denoted as $X_\sim$.*

2.2 Forward bisimulation

Now we study the notion of forward bisimulation which enables to lump the states of a weighted transition system.

Definition 2.4 (Forward bisimulation) *Let $(\mathcal{Q}, w)$ a weighted transition system and $\sim$ be an equivalence relation over the set $\mathcal{Q}$.*

The relation $\sim$ is called a forward bisimulation over the weighted transition system $(\mathcal{Q}, w)$ if and only if for every q, q', q'' such that $q \sim q'$, the following equation:

$$\textsc{flow}(\{q\}, [q'']_\sim) = \textsc{flow}(\{q'\}, [q'']_\sim)$$

is satisfied.

Question 1 ***(**)*** *Propose the largest forward bisimulation $\sim$ over the weighted transition systems depicted in Fig.* **??** *such that the state x is $\sim$-equivalent to no other state.*

That is to say that the relation $\sim$ should satisfy the following properties:

1. *$\sim$ is a forward bisimulation over the weighted transition system depicted in Fig.* **??***;*
2. *$[x]_\sim = \{x\}$;*
3. *for every two states $q, q' \in \mathcal{Q}$ and for every forward bisimulation $\sim'$ over the weighted transition system depicted in Fig.* **??** *such that $[x]_{\sim'} = \{x\}$, we have $q \sim' q' \Rightarrow q \sim q'$.*

Answer:

1. Let us show that the relation $\sim$ among the elements of $\mathcal{Q}$ that is defined as follows:

   $$\mathcal{Q}_\sim \triangleq \{\{x\}, \{y_1, y_2\}, \{y_3\}, \{z_1, z_2, z_3\}\}$$

   is a forward bisimulation.

   We compute in the following matrix the flow from every state in $\mathcal{Q}$ to every $\sim$-equivalence class in $\mathcal{Q}_\sim$.

FLOW	$\{x\}$	$\{y_1, y_2\}$	$\{y_3\}$	$\{z_1, z_2, z_3\}$
$\{x\}$	0	$\frac{2}{3}$	$\frac{1}{3}$	0
$\{y_1\}$	0	$\frac{1}{2}$	0	$\frac{1}{2}$
$\{y_2\}$	0	$\frac{1}{2}$	0	$\frac{1}{2}$
$\{y_3\}$	0	0	0	1
$\{z_1\}$	$\frac{1}{2}$	0	0	$\frac{1}{2}$
$\{z_2\}$	$\frac{1}{2}$	0	0	$\frac{1}{2}$
$\{z_3\}$	$\frac{1}{2}$	0	0	$\frac{1}{2}$

   We notice that the rows y_1 and y_2 are the same. Moreover, the three rows z_1, z_2, and z_3 are equal as well. Thus $\sim$ is a forward bisimulation.

2. Conversely, let $\sim$ be the largest forward bisimulation such that $[x]_\sim = \{x\}$.

   We have, for every $i \in \{1, 2, 3\}$,

   $$\textsc{flow}(y_i, [x]_\sim) = 0.$$

and for every $j \in \{1, 2, 3\}$,

$$\textsc{flow}(z_i, [x]_\sim) = \frac{1}{2}.$$

Thus, for every $i, j \in \{1, 2, 3\}$, we have $y_i \not\sim z_j$.

So, for every $i \in \{1, 2, 3\}$, $[y_i]_\sim \subseteq \{y_1, y_2, y_3\}$ and $[z_i]_\sim \subseteq \{z_1, z_2, z_3\}$.

Then, we have:

(a) $\textsc{flow}(y_1, [z_3]_\sim) \in \{0, \frac{1}{2}\}$;

(b) $\textsc{flow}(y_2, [z_3]_\sim) \in \{0, \frac{1}{2}\}$;

(c) $\textsc{flow}(y_3, [z_3]_\sim) = 1$;

Thus $\textsc{flow}(y_3, [z_3]_\sim) \neq \textsc{flow}(y_1, [z_3]_\sim)$ and $\textsc{flow}(y_3, [z_3]_\sim) \neq \textsc{flow}(y_2, [z_3]_\sim)$.

We can conclude that $[y_3]_\sim = \{y_3\}$ and that for every $i \in \{1, 2\}$, $[y_i]_\sim \subseteq \{y_1, y_2\}$.

It follows that the equivalence relation that is defined by the following equivalence classes:

$$\{\{x\}, \{y_1, y_2\}, \{y_3\}, \{z_1, z_2, z_3\}\}$$

is the largest forward bisimulation such that $\{x\}$ is an equivalence class.

Question 2 ***(*)*** *Let $(\mathcal{Q}, w)$ a weighted transition system and $\sim$ a forward bisimulation over $\mathcal{Q}$. Show that there exists a weighted transition system $(\mathcal{Q}^\sharp, w^\sharp)$ such that both following properties are satistified:*

1. *the states of the new weighted transition system are the $\sim$-equivalence class of the initial one (i.e. $\mathcal{Q}^\sharp = [\mathcal{Q}]_\sim$);*
2. *for every trace $\tau^\sharp = (C_i)_{0 \leqslant i \leqslant n}$ in the new weighted transition system and any initial state $q^\star \in C_0$, the probability (in the new weighted transition system) of the trace $\tau^\sharp$ is equal to the sum of the probabilities (in the former weighted transition system) of the traces $(q_i)_{0 \leqslant i \leqslant n}$ such that $q_0 = q^\star$ and $q_i \in C_i$ for every i between 1 and n.*

Answer:

We define $w^\sharp([q]_\sim, [q'']_\sim) \triangleq \textsc{flow}(\{q\}, [q'']_\sim)$, for every two states $q, q'' \in \mathcal{Q}$.
The function $w^\sharp$ is well defined, since for every $q, q', q'' \in \mathcal{Q}$, the following condition:

$$\textsc{flow}(\{q\}, [q'']_\sim) = \textsc{flow}(\{q'\}, [q'']_\sim)$$

is satisfied.
Moreover, for every state $q \in \mathcal{Q}$, we have:

$$\begin{array}{rcl} \sum_{C \in \mathcal{Q}_\sim} w^\sharp([q]_\sim, C) & = & \sum_{C \in \mathcal{Q}_\sim} \textsc{flow}(\{q\}, C) \\ & = & \textsc{flow}(\{q\}, \mathcal{Q}) \\ & = & 1. \end{array}$$

We prove the relationship over the probabilities of traces by induction.

1. The probability of the trace (C_0) is equal to 1.

   The probability of the trace $(q^\star)$ is equal to 1 as well.

2. We assume that the relationship holds for traces of size n.

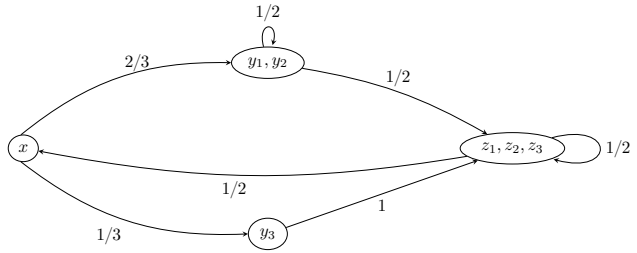
   Let $\tau^\sharp \triangleq (C_i)_{0\leqslant i\leqslant n+1}$ be a trace in the new transition system.

   By induction hypothesis, we assume the probability of the trace $(C_i)_{0\leqslant i\leqslant n}$ is equal to the sum of the probabilities of the traces $(q_i)_{0\leqslant i\leqslant n}$, in the initial transition system, such that $q_0 = q^\star$ and $q_i \in C_i$ for every i between 1 and n.

   We have:

$$\begin{array}{rcl} P(\tau^\sharp \mid C_0) & = & P((C_i)_{0\leqslant i\leqslant n} \mid C_0) \cdot w^\sharp(C_n, C_{n+1}) \\ & = & \left(\sum_{(q_i)_{0\leqslant i\leqslant n}, q_0=q^\star, q_i\in C_i} P((q_i)_{0\leqslant i\leqslant n} \mid q^\star)\right) \cdot w^\sharp(C_n, C_{n+1}) \\ & = & \sum_{(q_i)_{0\leqslant i\leqslant n}, q_0=q^\star, q_i\in C_i} P((q_i)_{0\leqslant i\leqslant n} \mid q^\star) \cdot w^\sharp(C_n, C_{n+1}) \\ & = & \sum_{(q_i)_{0\leqslant i\leqslant n}, q_0=q^\star, q_i\in C_i} P((q_i)_{0\leqslant i\leqslant n} \mid q^\star) \cdot \sum_{q_{n+1}\in C_{n+1}} w(q_n, q_{n+1}) \\ & = & \sum_{(q_i)_{0\leqslant i\leqslant n+1}, q_0=q^\star, q_i\in C_i} P((q_i)_{0\leqslant i\leqslant n+1} \mid q^\star) \end{array}$$

In our case study, we obtain the following coarse-grained transition system:



2.3 Backward bisimulation

Now we study the notion of backward bisimulation which highlights statistical invariants about the time evolution of the state distribution of the underlying weighted transition system.

Definition 2.5 (Backward bisimulation) *Let $(\mathcal{Q}, w)$ a weighted transition system and $\sim$ be an equivalence relation over the set $\mathcal{Q}$.*

The relation $\sim$ is called a backward bisimulation if and only if for every q, q', q'' such that $q \sim q'$, the following equation:

$$\textsc{flow}([q'']_\sim, \{q\}) = \textsc{flow}([q'']_\sim, \{q'\})$$

is satisfied.

Question 3 ***(**)*** *Propose the largest backward bisimulation $\sim$ over the weighted transition systems that is depicted in Fig.* **??**.

That is to say that the relation $\sim$ should satisfy both following properties:

1. *$\sim$ is a backward bisimulation over the weighted transition system depicted in Fig.* **??***;*
2. *for every states q and q' and for every backward bisimulation $\sim'$ over the weighted transition system depicted in Fig.* **??***, we have $q \sim' q' \Rightarrow q \sim q'$.*

Answer:

1. Let us show that the relation $\sim$ among the elements of $\mathcal{Q}$ that is defined as follows:

$$\mathcal{Q}_\sim \triangleq \{\{x\}, \{y_1, y_2\}, \{y_3\}, \{z_1, z_2\}, \{z_3\}\}$$

   is a backward bisimulation.

   We compute in the following matrix the flow from every $\sim$-equivalence class in $\mathcal{Q}_\sim$ to every state in $\mathcal{Q}$.

FLOW	$\{x\}$	$\{y_1\}$	$\{y_2\}$	$\{y_3\}$	$\{z_1\}$	$\{z_2\}$	$\{z_3\}$
$\{x\}$	0	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$	0	0	0
$\{y_1, y_2\}$	0	$\frac{1}{2}$	$\frac{1}{2}$	0	$\frac{1}{2}$	$\frac{1}{2}$	0
$\{y_3\}$	0	0	0	0	0	0	1
$\{z_1, z_2\}$	1	0	0	0	$\frac{1}{2}$	$\frac{1}{2}$	0
$\{z_3\}$	$\frac{1}{2}$	0	0	0	0	0	$\frac{1}{2}$

   We notice that the columns y_1 and y_2 are the same.

   Moreover, the columns z_1 and z_2 are equal as well.

   Thus $\sim$ is a backward bisimulation.

2. Conversely, let $\sim$ be the largest backward bisimulation over the weighted transition system that is depicted in Fig. **??**.

   Since $\mathcal{Q}_\sim$ is a partition of $\mathcal{Q}$, we get that:

$$\begin{array}{rcl} \textsc{flow}(\mathcal{Q}, \{q\}) & = & \sum_{C\in\mathcal{Q}_\sim} \textsc{flow}(\mathcal{Q}, \{q\}) \\ & = & \sum_{C\in\mathcal{Q}_\sim} \textsc{flow}(\mathcal{Q}, \{q'\}) \\ & = & \textsc{flow}(\mathcal{Q}, \{q'\}). \end{array}$$

   We have:

$$\begin{cases} \textsc{flow}(\mathcal{Q}, \{x\}) = \frac{3}{2}; \\ \textsc{flow}(\mathcal{Q}, \{y_1\}) = \frac{5}{6}; \\ \textsc{flow}(\mathcal{Q}, \{y_2\}) = \frac{5}{6}; \\ \textsc{flow}(\mathcal{Q}, \{y_3\}) = \frac{1}{3}; \\ \textsc{flow}(\mathcal{Q}, \{z_1\}) = 1; \\ \textsc{flow}(\mathcal{Q}, \{z_2\}) = 1; \\ \textsc{flow}(\mathcal{Q}, \{z_3\}) = \frac{3}{2}; \end{cases}$$

   It follows that:

   - $[x]_\sim \subseteq \{x, z_3\}$;

- $[y_1]_\sim \subseteq \{y_1, y_2\}$;
- $[y_2]_\sim \subseteq \{y_1, y_2\}$;
- $[y_3]_\sim = \{y_3\}$.
- $[z_1]_\sim \subseteq \{z_1, z_2\}$;
- $[z_2]_\sim \subseteq \{z_1, z_2\}$;
- $[z_3]_\sim \subseteq \{x, z_3\}$.

Since $\{y_3\}$ is a $\sim$-equivalence class, we have, for every two states q, q' such that $q \sim q'$:

$$\textsc{flow}(\{y_3\}, \{q\}) = \textsc{flow}(\{y_3\}, \{q'\}).$$

Since $\textsc{flow}(\{y_3\}, \{x\}) = 0$ and $\textsc{flow}(\{y_3\}, \{z_3\}) = 1$, it follows that $x \not\sim z_3$.

It follows that the equivalence relation that is defined by the following equivalence classes:

$$\{\{x\}, \{y_1, y_2\}, \{y_3\}, \{z_1, z_2\}, \{z_3\}\}$$

is the largest backward bisimulation for the weigthed transition system that is depicted in Fig. **??**.

Question 4 ***(***)*** *Let $(\mathcal{Q}, w)$ a weighted transition system and $\sim$ a backward bisimulation over $\mathcal{Q}$.*

Let $q^\star$ be a state such that $[q^\star]_\sim = \{q^\star\}$ and n be a natural number.

Show that for every two states $q, q' \in \mathcal{Q}$ such that $q \sim q'$, the probability that the system ends in state q after n computation steps knowing that it has started in state $q^\star$ is equal to the probability that the system ends in that q' after n computations knowing that it has started in state $q^\star$.

That is to say that the sum of the probabilities of all the traces of n transitions starting in state $q^\star$ and ending in state q is equal to the sum of the probabilities of all the traces of n transitions starting in state $q^\star$ and ending in state q'.

Answer:

We denote as $\mathcal{T}(q, n, q')$ the set of traces starting from the state q and ending in the state q' in exactly n transitions.

We prove the result by induction over n.

1. For $n = 0$, $\mathcal{T}(q^\star, n, q)$ is equal to $\{q^\star\}$ whenever $q = q^\star$ and to the empty set $\varnothing$ otherwise.
   Let $q, q' \in \mathcal{Q}$ be two states such that $q \sim q'$.
   (a) Whenever $q = q^\star$.
   Since $q^\star$ is the only element of its $\sim$-equivalence class, we have $q' = q^\star$ as well.
   Hence $\sum_{\tau \in \mathcal{T}(q^\star, n, q)} P(\tau \mid q^\star) = 1$ and $\sum_{\tau \in \mathcal{T}(q^\star, n, q')} P(\tau \mid q^\star) = 1$.
   (b) Whenever $q' = q^\star$.
   See previous case.
   (c) Otherwise.
   Both sums $\sum_{\tau \in \mathcal{T}(q^\star, n, q)} P(\tau \mid q^\star)$ and $\sum_{\tau \in \mathcal{T}(q^\star, n, q)} P(\tau \mid q^\star)$ are equal to 0.
   In every case, we have
   $$\sum_{\tau \in \mathcal{T}(q^\star, n, q)} P(\tau \mid q^\star) = \sum_{\tau \in \mathcal{T}(q^\star, n, q)} P(\tau \mid q^\star)$$

2. We assume that the property holds for $n \in \mathbb{N}$, let us show that it holds for the traces of length $n + 1$.
   Let q, q' be two states in $\mathcal{Q}$ such that $q \sim q'$.
   For every $\sim$-equivance class C, we choose q_C an element of C.
   We have:

$$\begin{array}{rcl}
\sum_{\tau \in \mathcal{T}(q^\star, n+1, q)} P(\tau \mid q^\star) & = & \sum_{q'' \in \mathcal{Q}, \tau' \in \mathcal{T}(q^\star, n, q'')} P(\tau' \mid q^\star) \cdot w(q'', q) \\
& = & \sum_{C \in \mathcal{Q}_\sim} \sum_{q'' \in C, \tau' \in \mathcal{T}(q^\star, n, q'')} P(\tau' \mid q^\star) \cdot w(q'', q) \\
& = & \sum_{C \in \mathcal{Q}_\sim} \sum_{q'' \in C} \sum_{\tau' \in \mathcal{T}(q^\star, n, q'')} P(\tau' \mid q^\star) \cdot w(q'', q) \\
& = & \sum_{C \in \mathcal{Q}_\sim} \sum_{q'' \in C} w(q'', q) \cdot \left(\sum_{\tau' \in \mathcal{T}(q^\star, n, q'')} P(\tau' \mid q^\star) \right) \\
& = & \sum_{C \in \mathcal{Q}_\sim} \sum_{q'' \in C} w(q'', q) \cdot \left(\sum_{\tau' \in \mathcal{T}(q^\star, n, q_C)} P(\tau' \mid q^\star) \right) \\
& = & \sum_{C \in \mathcal{Q}_\sim} \left(\sum_{q'' \in C} w(q'', q) \right) \cdot \left(\sum_{\tau' \in \mathcal{T}(q^\star, n, q_C)} P(\tau' \mid q^\star) \right) \\
& = & \sum_{C \in \mathcal{Q}_\sim} (\textsc{flow}(C, q)) \cdot \left(\sum_{\tau' \in \mathcal{T}(q^\star, n, q_C)} P(\tau' \mid q^\star) \right) \\
& = & \sum_{C \in \mathcal{Q}_\sim} (\textsc{flow}(C, q')) \cdot \left(\sum_{\tau' \in \mathcal{T}(q^\star, n, q_C)} P(\tau' \mid q^\star) \right) \\
& = & \sum_{C \in \mathcal{Q}_\sim} \left(\sum_{q'' \in C} w(q'', q') \right) \cdot \left(\sum_{\tau' \in \mathcal{T}(q^\star, n, q_C)} P(\tau' \mid q^\star) \right) \\
& = & \sum_{C \in \mathcal{Q}_\sim} \sum_{q'' \in C} w(q'', q') \cdot \left(\sum_{\tau' \in \mathcal{T}(q^\star, n, q_C)} P(\tau' \mid q^\star) \right) \\
& = & \sum_{C \in \mathcal{Q}_\sim} \sum_{q'' \in C} w(q'', q') \cdot \left(\sum_{\tau' \in \mathcal{T}(q^\star, n, q'')} P(\tau' \mid q^\star) \right) \\
& = & \sum_{C \in \mathcal{Q}_\sim} \sum_{q'' \in C} \sum_{\tau' \in \mathcal{T}(q^\star, n, q'')} P(\tau' \mid q^\star) \cdot w(q'', q) \\
& = & \sum_{C \in \mathcal{Q}_\sim} \sum_{q'' \in C, \tau' \in \mathcal{T}(q^\star, n, q'')} P(\tau' \mid q^\star) \cdot w(q'', q') \\
& = & \sum_{q'' \in \mathcal{Q}, \tau' \in \mathcal{T}(q^\star, n, q'')} P(\tau' \mid q^\star) \cdot w(q'', q') \\
& = & \sum_{\tau \in \mathcal{T}(q^\star, n+1, q')} P(\tau \mid q^\star)
\end{array}$$

   Thus, for every $n \in \mathbb{N}$ and every two $\sim$-equivalent states q, q' in $\mathcal{Q}$, we have:
   $$\sum_{\tau \in \mathcal{T}(q^\star, n, q)} P(\tau \mid q^\star) = \sum_{\tau \in \mathcal{T}(q^\star, n, q')} P(\tau \mid q^\star)$$

Question 5 ***(**)*** *Let $(\mathcal{Q}, w)$ a weighted transition system and $\sim$ a backward bisimulation over $\mathcal{Q}$.*
Let $q^\star$ be a state such that $[q^\star]_\sim = \{q^\star\}$.
Show that there exists a weighted transition system $(\mathcal{Q}^\sharp, w^\sharp)$ such that

1. *the states of the new weighted transition system are the $\sim$-equivalence class of the initial one (i.e. $\mathcal{Q}^\sharp = [\mathcal{Q}]_\sim$)*
2. *For every trace $\tau^\sharp = (C_i)_{0 \leqslant i \leqslant n}$ in the new weighted transition system such that $C_0 = \{q^\star\}$, the probability (in the new weighted transition system) of the trace $\tau^\sharp$ is equal to the sum of the probabilities (in the former weighted transition system) of the traces $(q_i)_{0 \leqslant i \leqslant n}$ such that $q_i \in C_i$ for every i between 0 and n.*

Answer:

We define $w^\sharp([q]_\sim, [q'']_\sim) \triangleq \frac{\sum_{q' \in [q]_\sim} \textsc{flow}(\{q'\}, [q'']_\sim)}{\textsc{Cardinal}([q]_\sim)}$, for every two states $q, q'' \in \mathcal{Q}$.

The function $w^\sharp$ is well defined, since for every $q, q', q'' \in \mathcal{Q}$, the following condition:

$$\textsc{flow}(\{q\}, [q'']_\sim) = \textsc{flow}(\{q'\}, [q'']_\sim)$$

is satisfied.

Moreover, for every state $q \in \mathcal{Q}$, we have:

$$\begin{aligned}
\sum_{C \in \mathcal{Q}_\sim} w^\sharp([q]_\sim, C) &= \sum_{C \in \mathcal{Q}_\sim} \frac{\sum_{q' \in [q]_\sim} \textsc{flow}(\{q'\}, C)}{\textsc{Cardinal}([q]_\sim)} \\
&= \frac{\sum_{C \in \mathcal{Q}_\sim} \sum_{q' \in [q]_\sim} \textsc{flow}(\{q'\}, C)}{\textsc{Cardinal}([q]_\sim)} \\
&= \frac{\sum_{q' \in [q]_\sim} \textsc{flow}(\{q'\}, \mathcal{Q})}{\textsc{Cardinal}([q]_\sim)} \\
&= \frac{\sum_{q' \in [q]_\sim} 1}{\textsc{Cardinal}([q]_\sim)} \\
&= 1
\end{aligned}$$

We prove the relationship over the probabilities of traces by induction.

1. The probability of the trace (C_0) is equal to 1.

   The probability of the trace $(q^\star)$ is equal to 1 as well.

2. We assume that the relationship holds for traces of size n.

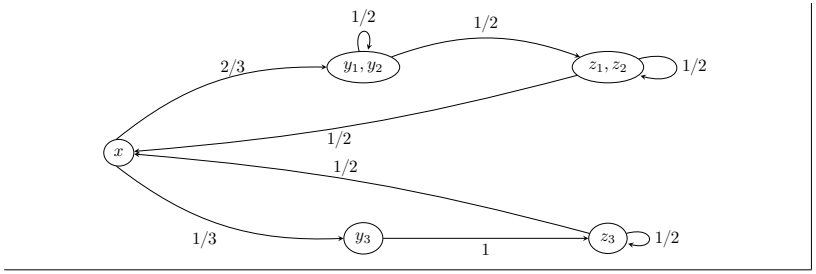
   Let $\tau^\sharp \triangleq (C_i)_{0 \leqslant i \leqslant n+1}$ be a trace in the new transition system.

   By induction hypothesis, we assume the probability of the trace $(C_i)_{0 \leqslant i \leqslant n}$ is equal to the sum of the probabilities of the traces $(q_i)_{0 \leqslant i \leqslant n}$, in the initial transition system, such that $q_0 = q^\star$ and $q_i \in C_i$ for every i between 1 and n.

   We have:

$$\begin{aligned}
P(\tau^\sharp \mid C_0) &= P((C_i)_{0 \leqslant i \leqslant n} \mid C_0) \cdot w^\sharp(C_n, C_{n+1}) \\
&= \left(\sum_{(q_i)_{0 \leqslant i \leqslant n}, q_0 = q^\star, q_i \in C_i} P((q_i)_{0 \leqslant i \leqslant n} \mid q^\star)\right) \cdot w^\sharp(C_n, C_{n+1}) \\
&= \sum_{(q_i)_{0 \leqslant i \leqslant n}, q_0 = q^\star, q_i \in C_i} P((q_i)_{0 \leqslant i \leqslant n} \mid q^\star) \cdot w^\sharp(C_n, C_{n+1}) \\
&= \sum_{(q_i)_{0 \leqslant i \leqslant n}, q_0 = q^\star, q_i \in C_i} P((q_i)_{0 \leqslant i \leqslant n} \mid q^\star) \cdot \frac{\sum_{q'_n \in C_n} w(q'_n, C_{n+1})}{\textsc{Cardinal}(C_n)} \\
&= \sum_{(q_i)_{0 \leqslant i \leqslant n}, q_0 = q^\star, q_i \in C_i} P((q_i)_{0 \leqslant i \leqslant n} \mid q^\star) \cdot \frac{\sum_{q'_n \in C_n} w(q_n, C_{n+1})}{\textsc{Cardinal}(C_n)} \\
&= \sum_{(q_i)_{0 \leqslant i \leqslant n}, q_0 = q^\star, q_i \in C_i} P((q_i)_{0 \leqslant i \leqslant n} \mid q^\star) \cdot w(q_n, C_{n+1}) \frac{\sum_{q'_n \in C_n} 1}{\textsc{Cardinal}(C_n)} \\
&= \sum_{(q_i)_{0 \leqslant i \leqslant n}, q_0 = q^\star, q_i \in C_i} P((q_i)_{0 \leqslant i \leqslant n} \mid q^\star) \cdot w(q_n, C_{n+1}) \\
&= \sum_{(q_i)_{0 \leqslant i \leqslant n}, q_0 = q^\star, q_i \in C_i} \left(P((q_i)_{0 \leqslant i \leqslant n} \mid q^\star) \cdot \left(\sum_{q_{n+1} \in C_{n+1}} w(q_n, q_{n+1})\right)\right) \\
&= \sum_{(q_i)_{0 \leqslant i \leqslant n}, q_0 = q^\star, q_i \in C_i} \sum_{q_{n+1} \in C_{n+1}} P((q_i)_{0 \leqslant i \leqslant n} \mid q^\star) \cdot w(q_n, q_{n+1}) \\
&= \sum_{(q_i)_{0 \leqslant i \leqslant n}, q_0 = q^\star, q_i \in C_i} \sum_{q_{n+1} \in C_{n+1}} P((q_i)_{0 \leqslant i \leqslant n+1} \mid q^\star) \\
&= \sum_{(q_i)_{0 \leqslant i \leqslant n+1}, q_0 = q^\star, q_i \in C_i} P((q_i)_{0 \leqslant i \leqslant n+1} \mid q^\star)
\end{aligned}$$

In our case study, we obtain the following coarse-grained transition system:



3 Bisimulations induced by perfect symmetries among pairs of sites

Figure 2: A set of rules with two interaction sites having the same capabilities of interaction.

In this section, we consider the model that is made of the set of rules given in Fig. **??**. In this model, the role of sites r and l is intuitively the same. The goal of this section is to investigate what it means with respect to the set of rules and to extrapolate which bisimulations are induced by this property.

Definition 3.1 *The symmetric of an agent is obtained by swapping the states of the sites l and r of this agent.*

In particular:

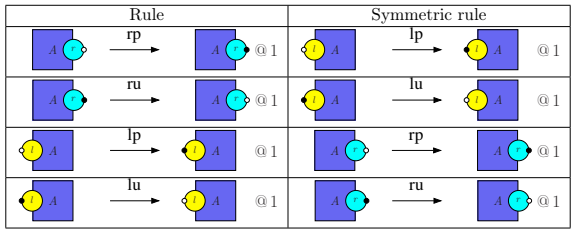
1. *If this agent contains neither the site l, nor the site r, the agent remains unchanged;*
2. *If this agent contains the site l, but not the site r, the site l is replaced with the site r;*
3. *If this agent contains the site r, but not the site l, the site r is replaced with the site l;*
4. *If this agent contains both the site l and the site r, the site l takes the former state of the site r while the site r takes the former state of the site l.*

Definition 3.2 *The symmetric of a rule is obtained by taking the symmetric of the left hand side and the symmetric of the right hand side.*

Question 6 ***(*)*** *Show that the symmetric of any rule of the model, is also a rule of the model with the same rate.*

Answer:

In the following array are drawn each rule (left column) and its symmetric (right column):

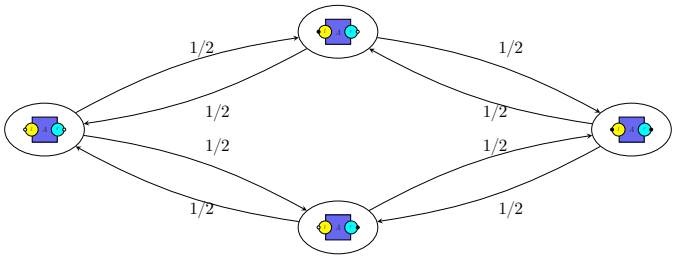


Question 7 *(*) Draw the weighted transition system that is induced by the rules of the model.*

So as to make this transition system easy to write, we consider only the states made of a single agent.

We recall that, an event e stemming from a state q is defined by a rule r and an embedding from the left hand side of the rule r into the state q. The propensity of the event e is equal to the rate of the rule. The state q' that is reached when applying the event is defined by the operational semantics of Kappa. Then the propability $w(q, q')$ is defined as the quotient between the sum of the propensities of the events from the state q to the state q' and the sum of the propensities of all the events stemming from the state q.

Answer:



Question 8 *(*) Show that the equivalence relation that gathers states by symmetry-classes induces both a forward bisimulation and a backward bisimulation.*

Answer:

1. Forward bisimulation:

FLOW			
	$\frac{1}{2}$	0	$\frac{1}{2}$
	$\frac{1}{2}$	0	$\frac{1}{2}$

2. Backward bisimulation:

FLOW		
	$\frac{1}{2}$	$\frac{1}{2}$
	0	0
	$\frac{1}{2}$	$\frac{1}{2}$

Thus this equivalence relation induces both a forward bisimulation and a backward bisimulation.

4 Bisimulations induced by contextual symmetries among pairs of sites

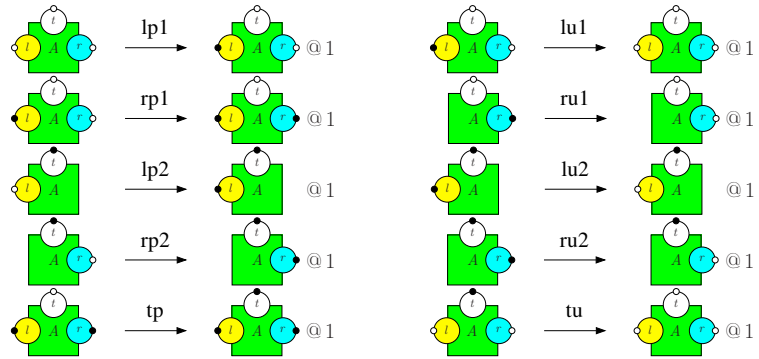


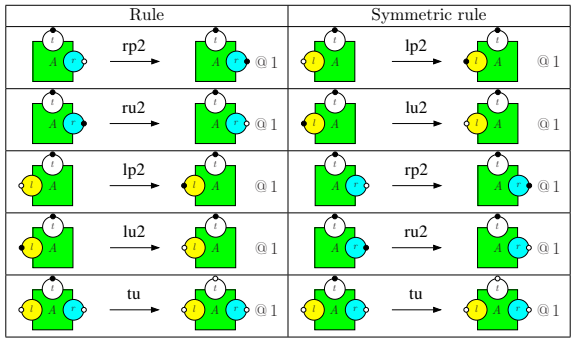
Figure 3: A set of rules in which two sites have the same capabilities of interactions only when a third site is activated.

In this section, we consider the model that is made of the set of rules given in Fig. **??**. In this model, the sites l and r get sequentially activated/deactivated when the site t is not activated, whereas they get activated/deactivated in parallel when this site is activated. Intuitively, the sites l and r have the same capabilities of interaction only when the site t is activated. Hence the symmetry between the site l and r is contextual. The goal of this section is to investigate whether contextual symmetries enjoy the same properties as uncontextual ones and to adapt the framework accordingly.

Question 9 *(*) Show that the symmetric of any rule of the model that requires the site t to be phosphorylated, is also a rule of the model with the same rate.*

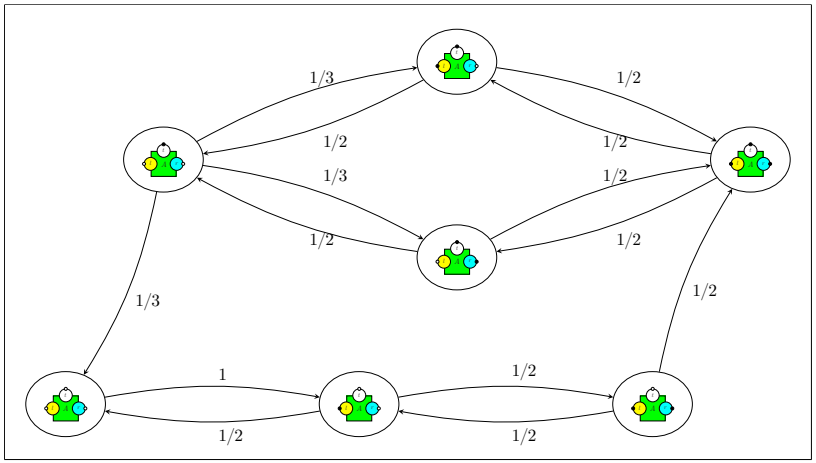
Answer:

In the following array, are drawn each rule (left column) and its symmetric (right column):



Question 10 *(*) Draw the weighted transition system that is induced by the rule of the model. So as to make this transition system easy to write, we consider the states made only of a single agent.*

Answer:



We consider the equivalence relation $\sim$ that identifies only the two following configurations:

(i.e. any other configuration is the only element of its equivalence class.)

Question 11 *(*) Show that the equivalence relation $\sim$ induces both a forward bisimulation and a backward bisimulation.*

Answer:

1. Forward bisimulation:

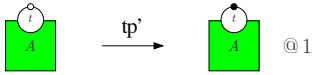
FLOW							
	0	0	0	0	$\frac{1}{2}$	0	$\frac{1}{2}$
	0	0	0	0	$\frac{1}{2}$	0	$\frac{1}{2}$

2. Backward bisimulation:

FLOW		
	0	0
	0	0
	0	0
	0	0
	$\frac{1}{3}$	$\frac{1}{3}$
	0	0
	$\frac{1}{2}$	$\frac{1}{2}$

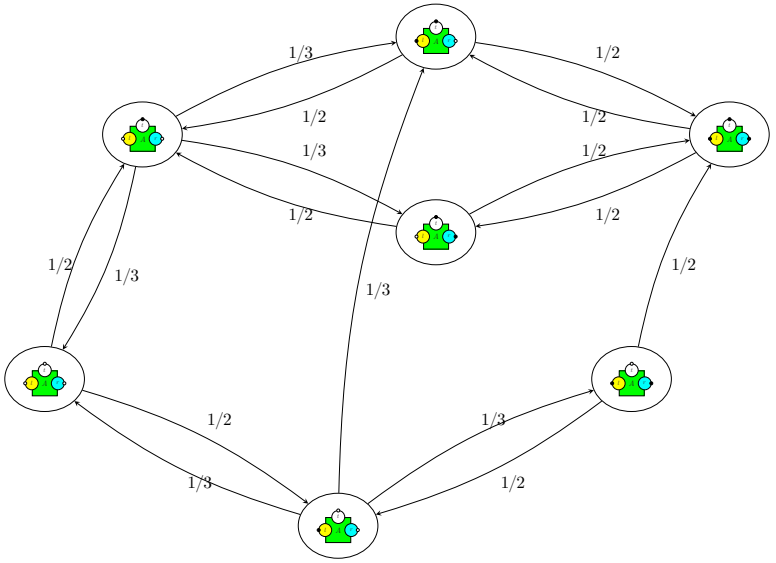
Thus, the relation $\sim$ is both a forward and a backward bisimulation.

Question 12 *(*) In the rules given in Fig.* **??**, *we propose to replace the rule tp by the following one:*



Is the relation $\sim$ still a forward bisimulation over the underlying weighted transition system?
Is the relation $\sim$ still a backward bisimulation over the underlying weighted transition system?

Answer:



1. Forward bisimulation:

FLOW							
	0	0	0	0	$\frac{1}{2}$	0	$\frac{1}{2}$
	0	0	0	0	$\frac{1}{2}$	0	$\frac{1}{2}$

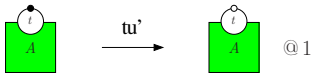
Thus the relation $\sim$ is a forward bisimulation.

2. Backward bisimulation:

FLOW		
	0	0
	$\frac{1}{3}$	0
	0	0
	$\frac{1}{3}$	$\frac{1}{3}$
	0	0
	$\frac{1}{2}$	$\frac{1}{2}$

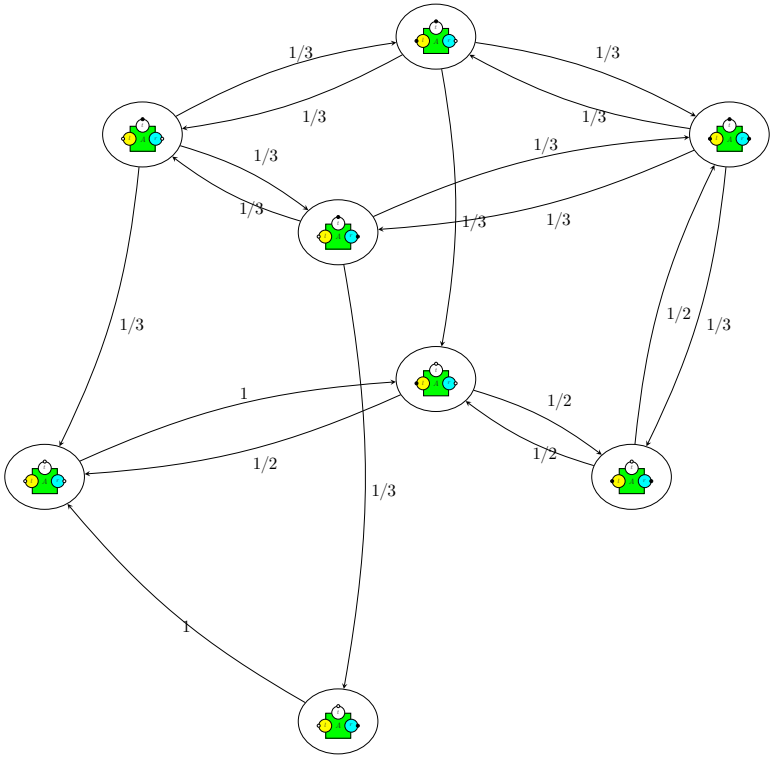
Thus the relation $\sim$ is not a backward bisimulation.

Question 13 ***(*)*** *In the rules given in Fig.* **??**, *we propose to replace the rule tu by the following one:*



Is the relation $\sim$ still a forward bisimulation over the underlying weighted transition system?
Is the relation $\sim$ still a backward bisimulation over the underlying weighted transition system?

Answer:



1. Forward bisimulation:

FLOW							
	0	$\frac{1}{3}$	0	0	$\frac{1}{3}$	0	$\frac{1}{3}$
	0	0	$\frac{1}{3}$	0	$\frac{1}{3}$	0	$\frac{1}{3}$

Thus, the relation $\sim$ is not a forward bisimulation.

2. Backward bisimulation:

FLOW		
	0	0
	0	0
	0	0
	0	0
	$\frac{1}{3}$	$\frac{1}{3}$
,	0	0
	$\frac{1}{3}$	$\frac{1}{3}$

Thus the relation $\sim$ is a backward bisimulation.

Question 14 ***(*****)*** *Propose a criterion over the rules and the state space of a Kappa model so as to ensure that a contextual symmetry among a pair of sites induces a forward bisimulation over the underlying weighted transition system.*

Answer:

To ensure that an equivalence relation induces a forward bisimulation, it is enough to prove that for every pair of symmetric states, any transition from the first state to another one can be mimicked by a transition of same rate from the second one, such that both targeted states are symmetric.

Thus, we require two conditions:

- For every rule r the lhs of which satisfies the contextual condition, the symmetric of the rule is a rule with the same rate.
- Every rule that can potentially break the contextual condition can only be applied on a symmetric configuration.

(The second condition ensures that any symmetry that is valid in the state of system before applying a rule is still valid after having applied this rule. While the first condition ensures that there is a rule to goes from the symmetric of the state before applying the rule to the symmetric of the state before applying the rule.)

Question 15 ***(*****)*** *Propose a criterion over the rules and the state space of a Kappa model so as to ensure that a contextual symmetry among a pair of sites induces a backward bisimulation over the underlying weighted transition system.*

Answer:

To ensure that an equivalence relation induces a backward bisimulation, it is enough to prove that for every pair of symmetric states, any transition ending in the first state can be mimicked by a transition of same rate ending in the second one, such that the sources of these both transitions are symmetric.

Thus, we require two conditions:

- For every rule r the lhs of which satisfies the contextual condition, the symmetric of the rule is a rule with the same rate.
- Every rule that can potentially forge the contextual condition can produce only in symmetric configurations.

(The second condition ensures that any symmetry that is valid in the state of system after applying a rule is still valid before having applied this rule. While the first condition ensures that there is a rule to goes from the symmetric of the state before applying the rule to the symmetric of the state before applying the rule.)

Master 2 MPRI course 2-19 Biochemical Programming
Jérôme Feret

Written examination 3 hours
All printed documents allowed
Any electronic devide prohibited
March 8th 2023

Abstract

We study the notion of contextual symmetry on the differential semantics of three variants of a model.

We consider a model with only one kind of agent and three sites which can be phosphorylated, or not. Each kind of site is identified by its position, (on the top, on the left, on the right). Unphosphorylated sites carry a white circular while phosphorylated ones carry a black one.

1 First variant of the model

The first variant of the model is described in Fig. 1.

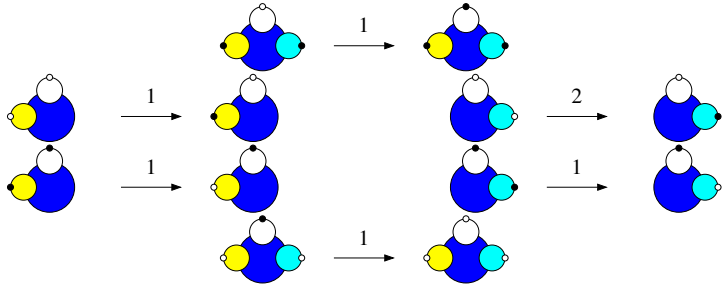


Figure 1: First case study.

It is worth noticing that when the site on the top is phosphorylated, the site on the left and the site on the right exhibit the same behavior in the sense that they share the same phosphorylation rate. We say that these sites are symmetric when the site on the top is phosphorylated. We call this a contextual symmetry.

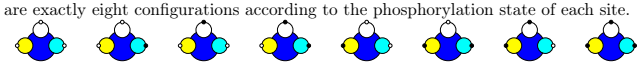
Our goal is to investigate the consequence of contextual symmetries on the behavior of models.

Question 1 (Configuration space) *Enumerate all the configurations the protein can take ?*

We denote by $\mathcal{V}$ the set of the configurations of the protein.

Answer:

There are exactly eight configurations according to the phosphorylation state of each site.



Question 2 (Differential semantics) *Write the system of ordinary differential equations that describes the evolution of the concentration of each potential configuration of the protein.*

This system takes the form:

$$\frac{d\vec{X}(t)}{dt} = \mathbb{F}(\vec{X}(t))$$

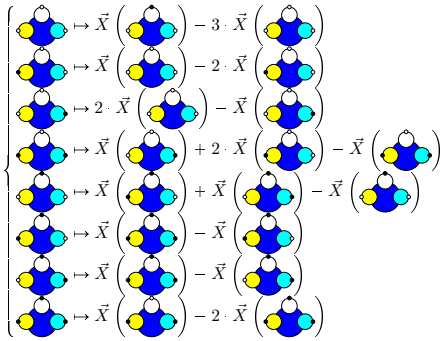
where $\vec{X}(t)$ is the function mapping each configuration $x \in \mathcal{V}$ of the protein to its concentration at time t and $\mathbb{F}$ is a function from $\mathbb{R}^{\mathcal{V}}$ into itself.

Answer:

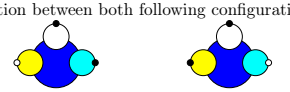
By applying mass action principle, we obtain the following system of differential equations:

$$\frac{\mathrm{d}\vec{X}(t)}{\mathrm{d}t} = \mathbb{F}(\vec{X}(t))$$

where: $\mathbb{F}(\vec{X}) =$



We propose to ignore the distinction between both following configurations of the protein:

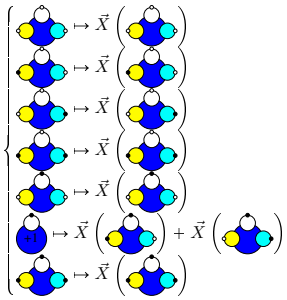


which comes down to replace the variables standing for the concentration of these configurations with a single one standing for the sum of their values.

Question 3 (Abstraction) *Introduce a set of abstract observables $\mathcal{V}^\sharp$ and a linear function ϕ from the set $\mathbb{R}^{\mathcal{V}}$ into the set $\mathbb{R}^{\mathcal{V}^\sharp}$ to model this change of variables.*

Answer:

$\phi(\vec{X}) =$



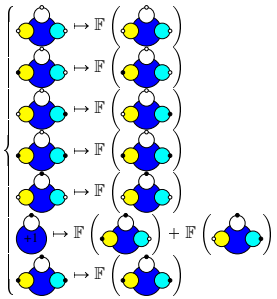
We say that ϕ induces a forward bisimulation is there exists a function $\mathbb{F}^\sharp$ from the set $\mathbb{R}^{\mathcal{V}^\sharp}$ into itself such that the property $\phi \circ \mathbb{F} = \mathbb{F}^\sharp \circ \phi$ is satisfied.

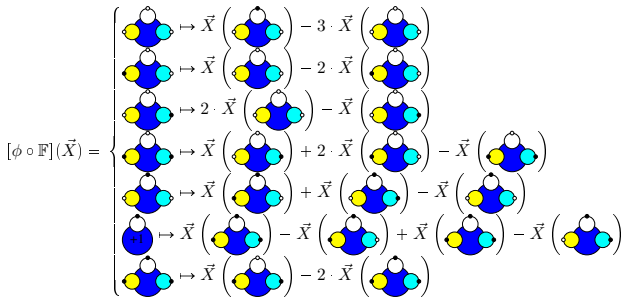
Question 4 (Forward bisimulation) *Does the function ϕ induce a forward bisimulation ? If so, express the corresponding function $\mathbb{F}^\sharp$.*

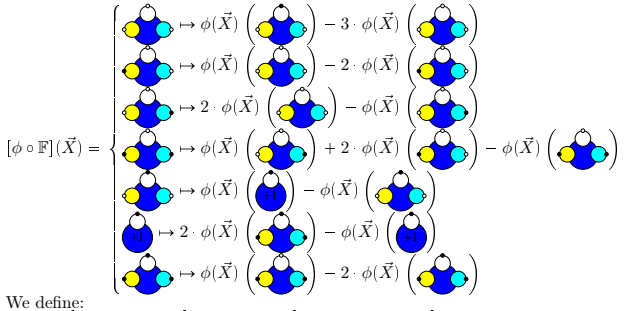
Answer:

We have:

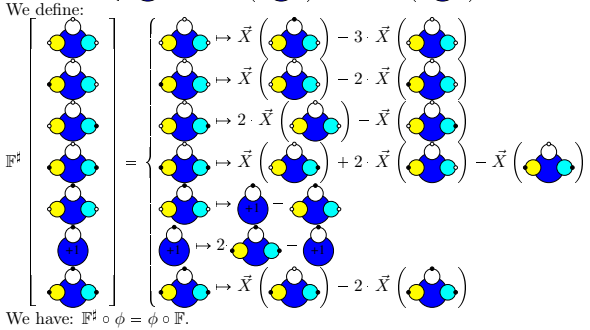
$[\phi \circ \mathbb{F}](\vec{X}) =$







We define:



We have: $\mathbb{F}^\sharp \circ \phi = \phi \circ \mathbb{F}$.

This proves that ϕ induces a forward bisimulation.

We now investigate about the potential relationships among the concentration of both following configurations:

We say that a pair of configurations induces a backward bisimulation if and only, the concentrations of these configurations remain equal for every solution of the differential semantics that starts in a state when the concentration of these configurations are equal.

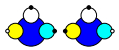
Question 5 (Backward bisimulation) *Does this pair of configurations induce a backward bisimulation?*

Answer:

We have:

Thus:

It follows that $\vec{X}\left(\quad\right) = \vec{X}\left(\quad\right)$ forever, provided that $\vec{X}_0\left(\quad\right) = \vec{X}_0\left(\quad\right)$.

Thus, the pair  induces a backward bisimulation.

2 A second variant of the model

We propose to relax the constraints on the phosphorylation of the site on the top. We obtain the second variant of the model which is described in Fig. 2.

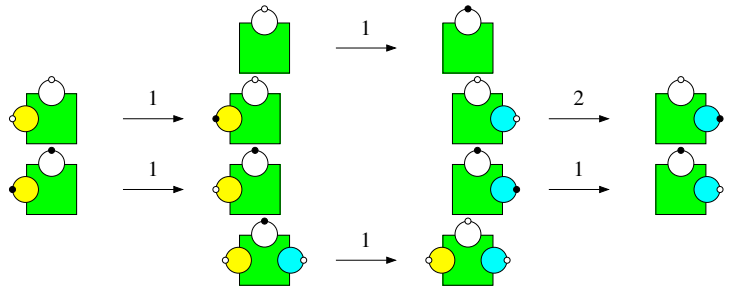


Figure 2: Second variant of the model.

Question 6 *Repeat questions 2, 4, and 5 to the variant of the model that is described in Fig. 2 with the two following configurations of interest:*

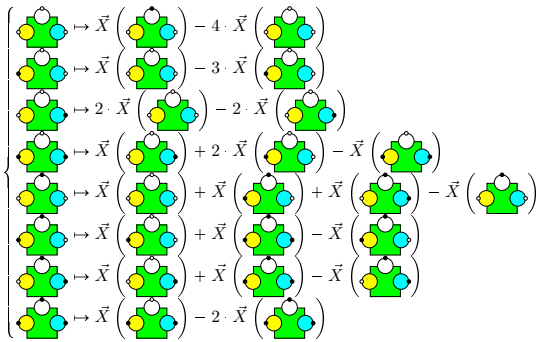
Answer:

1. There are exactly eight configurations according to the phosphorylation state of each site.

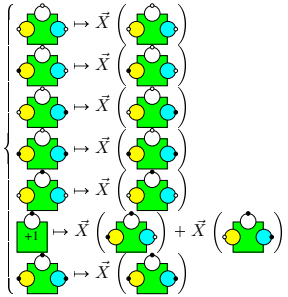
2. By applying mass action principle, we obtain the following system of differential equations:

$$\frac{\mathrm{d}\vec{X}(t)}{\mathrm{d}t} = \mathbb{F}(\vec{X}(t))$$

where: $\mathbb{F}(\vec{X}) =$

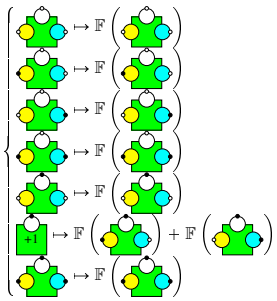


3. $\phi(\vec{X}) =$

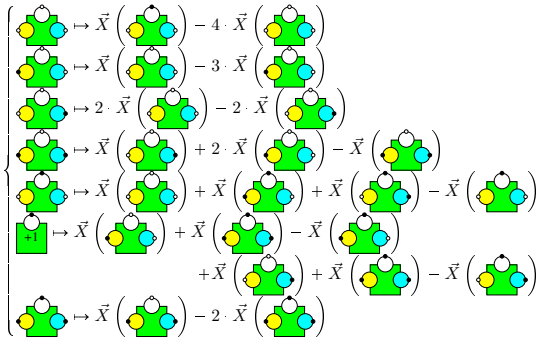


4. We have:

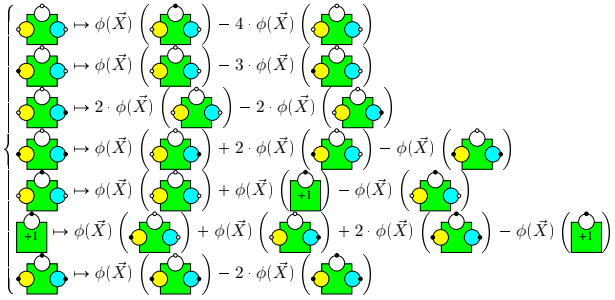
$[\phi \circ \mathbb{F}](\vec{X}) =$



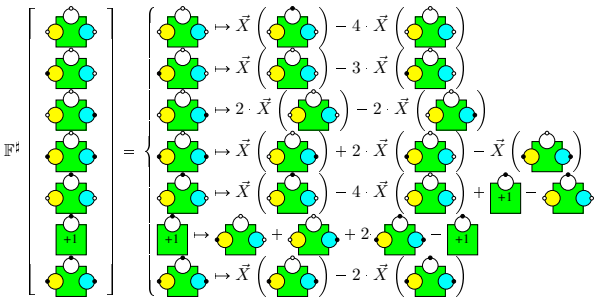
$[\phi \circ \mathbb{F}](\vec{X}) =$



$[\phi \circ \mathbb{F}](\vec{X}) =$

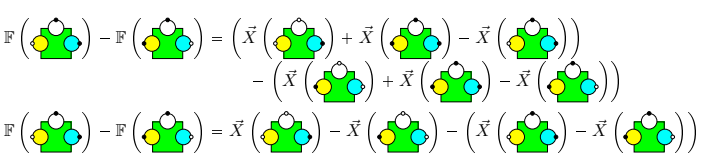


We define:

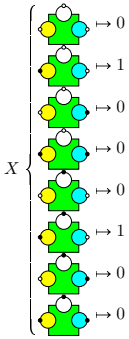


We have: $\mathbb{F}^\sharp \circ \phi = \phi \circ \mathbb{F}$.
This proves that ϕ induces a forward bisimulation.

5. We have:

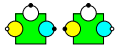


We consider the vector $\vec{X}$ that is defined as follows:



We have: $\vec{X}\left(\right) = \vec{X}\left(\right)$

But $\mathbb{F}\left(\right) - \mathbb{F}\left(\right) \neq 0$.

Thus, the pair of configurations  does not induce a backward bisimulation.

3 A third variant of the model

Now, we propose instead to relax the constraints on the dephosphorylation of the site on the top. This third variant is given in Fig. 3.

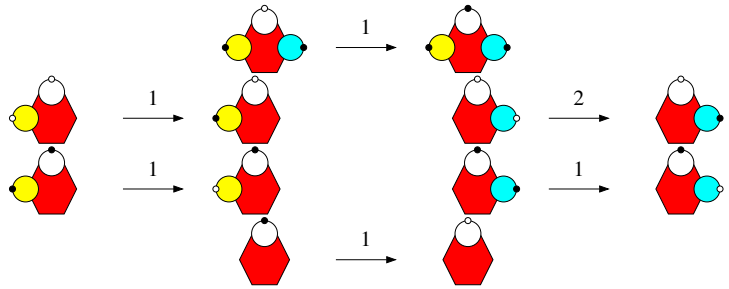


Figure 3: Third case study.

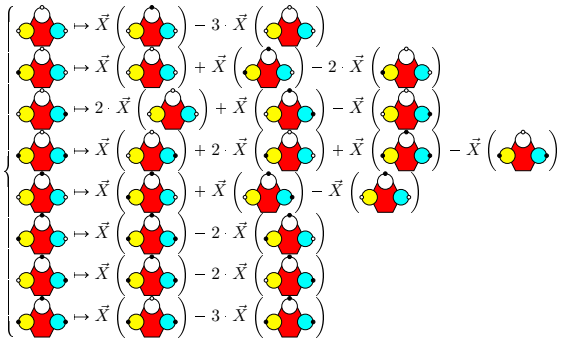
Question 7 *Repeat questions 2, 4, and 5 to the variant of the model that is described in Fig. 3 with the two following configurations of interest:*

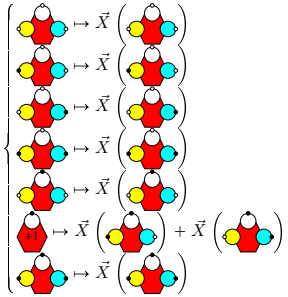
Answer:

1. There are exactly eight configurations according to the phosphorylation state of each site.

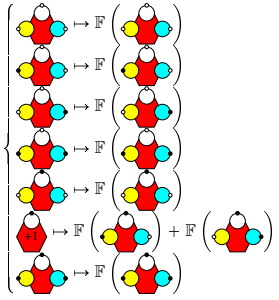
2. By applying mass action principle, we obtain the following system of differential equations:

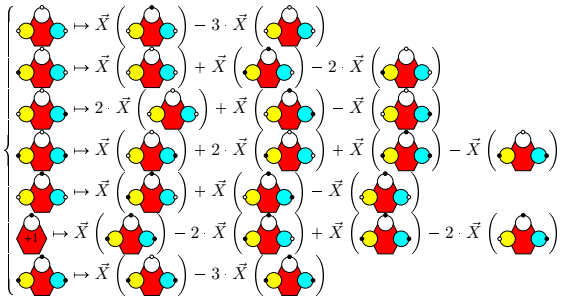
$$\frac{\mathrm{d}\vec{X}(t)}{\mathrm{d}t} = \mathbb{F}(\vec{X}(t))$$

where: $\mathbb{F}(\vec{X}) =$ 

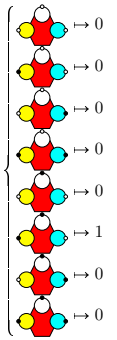
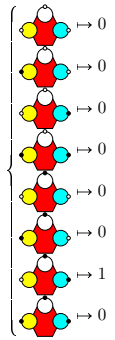
3. $\phi(\vec{X}) =$ 

4. We have:

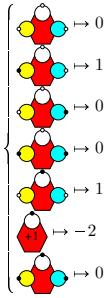
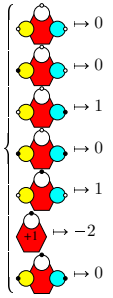
$[\phi \circ \mathbb{F}](\vec{X}) =$ 

$[\phi \circ \mathbb{F}](\vec{X}) =$ 

We take the following two vectors $\vec{X_1}$ and $\vec{X_2}$.

$\vec{X_1} =$  and $\vec{X_2} =$ 

We have: $\phi(\vec{X_1}) = \phi(\vec{X_2})$.

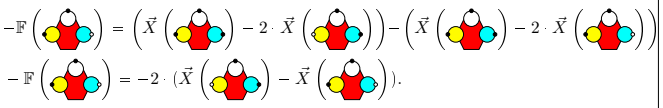
But: $\phi(\mathbb{F}(\vec{X_1})) =$  and $\phi(\mathbb{F}(\vec{X_2})) =$  .

So $\phi(\mathbb{F}(\vec{X_1})) \neq \phi(\mathbb{F}(\vec{X_2}))$.

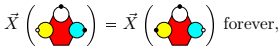
So there is no function $\mathbb{F}'$ such that $[\phi \circ \mathbb{F}] = [\mathbb{F}' \circ \phi]$.

This proves that ϕ does not induce a forward bisimulation.

5. We have:



Thus:

It follows that  forever, provided that .

Thus, the pair  induces a backward bisimulation.

4 Wrapping-up

Question 8 *Propose some sufficient conditions over the rules of a model to ensure that some contextual symmetries induce a forward bisimulation?*

Answer:

To ensure that some contextual symmetries induce a forward bisimulation, we must check that starting from two symmetric configurations, for any interaction that can be applied to the first one, there is an interaction (not necessarily the same one) that can be applied to the second one, and that leads to a symmetric configuration with the same kinetics.

1. In the two first variants of the model.

   The only interaction that apply to both symmetric configurations is the phosphorylation of the only remaining unphosphorylated site. This leads to the same configuration (with the three sites phosphorylated) at the same rate (1).

   This ensures that the pair of configurations induces a forward bisimulation.

2. In the third variant of the model.

   Starting for two symmetric configurations:

   we can dephosphorylate the top site at rate 1, to get the configuration: hfill 

But the configuration  has no other symmetric configuration and there is no way to transform the configuration  into  .

Thus, our sufficient condition is not satisfied.

Question 9 *Propose some sufficient conditions over the rules of a model to ensure that some contextual symmetries induce a backward bisimulation?*

Answer:

To ensure that some contextual symmetries induce a backward bisimulation, we must check that starting from two symmetric configurations, for any interaction that can be applied to get the first one, there is an interaction (not necessarily the same one) that can be applied to get the second one, and that starts from a symmetric configuration with the same kinetics.

1. In the first and third variant of the model.

   The only interactions to get symmetric configurations are the phosphorylations of the left or right site from the configuration where only the top site is phosphorylated. They start from the same configuration and have the same rate.

   This ensures that the pair of configurations induces a backward bisimulation.

2. In the second variant of the model.

   Considering the both following symmetric configurations:

   the configuration: hfill  can be obtained by phosphorylating the top site in the configuration  .

   But the configuration  has no other symmetric configuration and there is no way to transform the configuration  into  .

   Thus, our sufficient condition is not satisfied.

Master 2 MPRI C2-19 Biochemical Programming

Solution to Final Written Examination, March 8th 2023

Part II: On-line Analog Chemical Computation - François Fages

The subject of this examination is given in a BIOCHAM notebook because BIOCHAM will provide the corrected answers.

Practice

Let us consider the following chemical reaction network (CRN) in Biocham syntax, with mass action law kinetics with rate constant =1 by default.

This CRN can be considered as a simple signaling network in a cell relating

- some input species I, such as a ligand,
- to an output species O, such as the active form of some receptor R,
- itself produced by the cell and activated by ligand I.

In [1]:
```
I+R => I+O.
_ => R.
R => _.
O => _.

present(I,i). present(R,r). present(O, o).
parameter(i=1, r=1, o=0).
```

In [2]:
```
list_model.
```

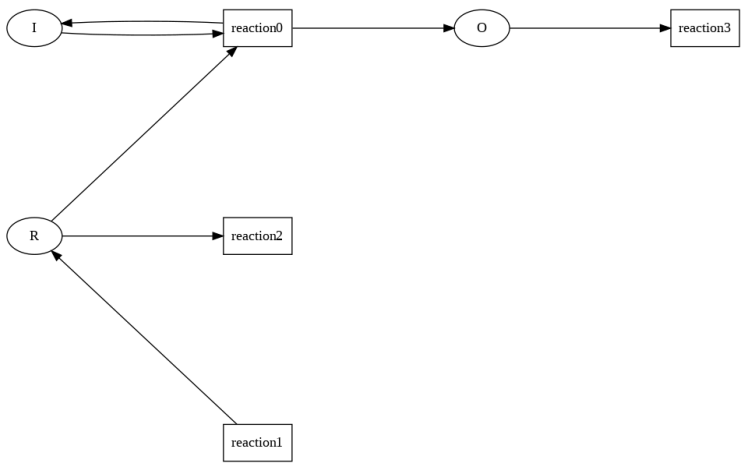
Out[2]:
```
MA(1) for I+R=>I+O.
MA(1) for _=>R.
MA(1) for R=>_.
MA(1) for O=>_.
initial_state(I=i).
initial_state(R=r).
initial_state(O=o).
parameter(
  i = 1,
  r = 1,
  o = 0
).
```

Question 1. (1 point)

Draw the reaction hypergraph of this CRN, in the form of a bipartite graph with species and reaction nodes.

Answer

In [3]:
```
draw_reactions.
```



Question 2. (1 point)

Write the ordinary differential equations associated to this CRN (still with reaction rate constants equal to 1)

Answer

In [4]:
```
list_ode.
```

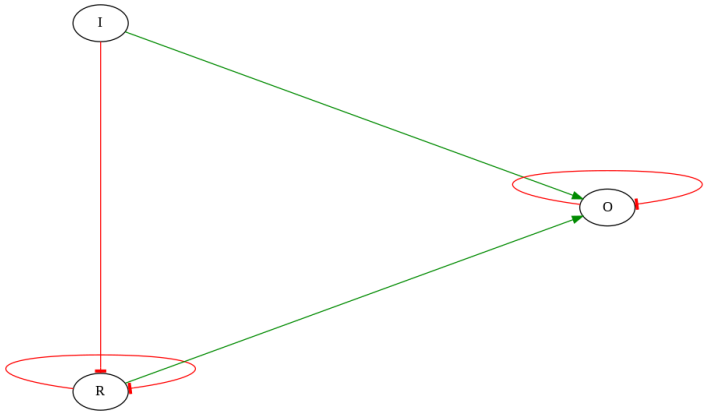
$$
\begin{aligned}
O_0 &= 0 \\
R_0 &= 1 \\
I_0 &= 1 \\
i &= 1 \\
r &= 1 \\
o &= 0 \\
\frac{dO}{dt} &= I * R - O \\
\frac{dR}{dt} &= 1 - R - I * R \\
\frac{dI}{dt} &= 0
\end{aligned}
$$

Question 3. (1 point)

Draw the influence graph of this CRN, i.e. the labeled graph of positive and negative influences between species.

Answer

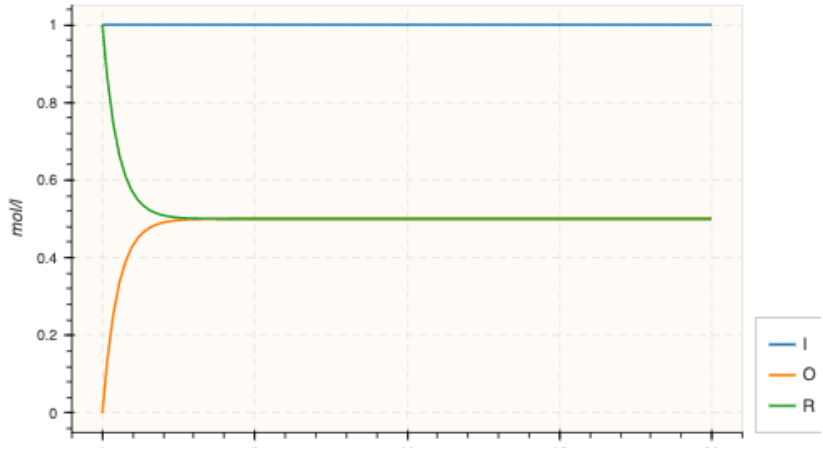
In [5]:
```
draw_influences.
```

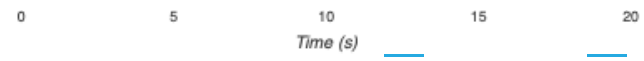


Simulations

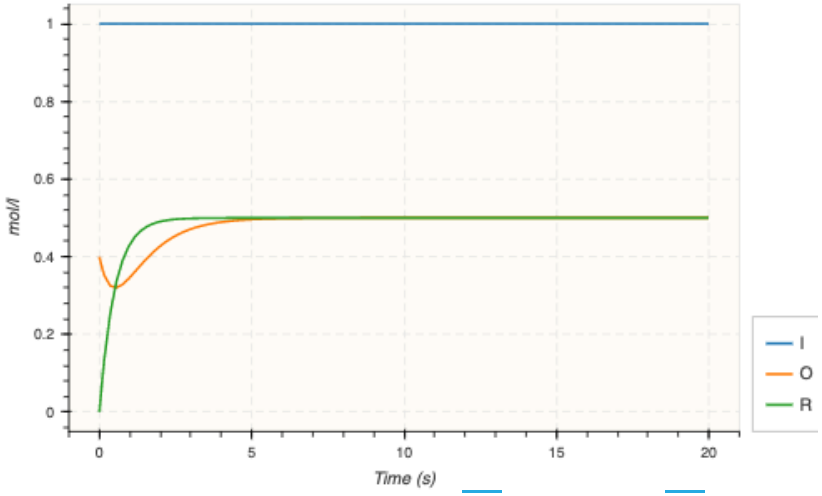
The following simulations suggest that this CRN computes a function of the input concentraiton I in the output concentration O, **independently of the initial concentrations of the other variables** R and O.

In [6]:
```
parameter(i=1, r=1, o=0). numerical_simulation. plot.
```

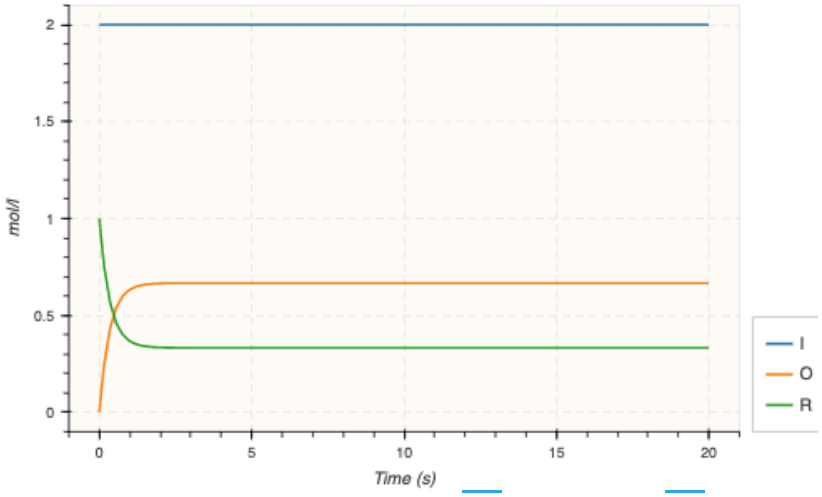




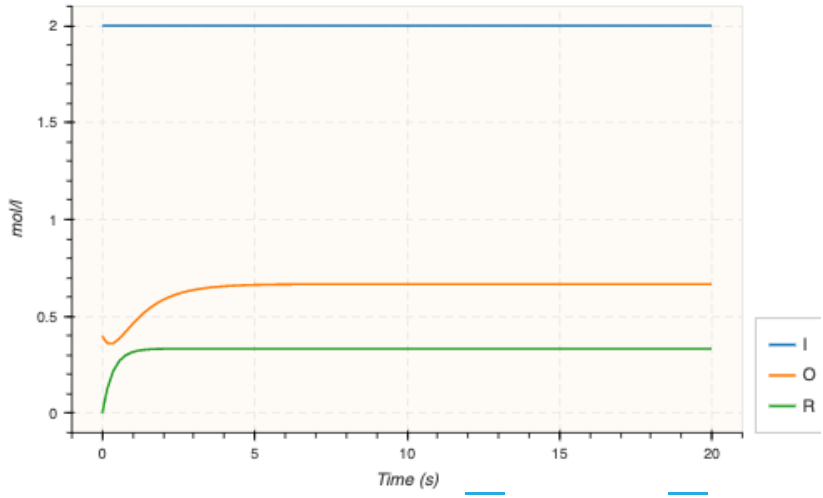
In [7]:
```
parameter(i=1, r=0, o=0.4). numerical_simulation. plot.
```



In [8]:
```
parameter(i=2, r=1, o=0). numerical_simulation. plot.
```



In [9]:
```
parameter(i=2, r=0, o=0.4). numerical_simulation. plot.
```



Question 4. (2 points)

Show that this system has a single steady state in the differential semantics.

Give the function computed by this CRN by writing the output concentration O as a function $O(I(0))$ of the input concentration I at steady state.

Answer

At steady state, all differential equations are zero.

By eliminating R = 1/(1+I) and I=I(0)

we get a unique solution O(I) = I/(1+I) = O(I(0))

Meaning

The previous CRN **computes an analog input/ouput function "on-line"** in the sense that the computed output variable O depends solely on the input variable I once it is pinned, i.e. stabilized at some value $I(0)$, whatever perturbations may have occured on the other variables R and O, i.e. for any values $R(0)$ and $O(0)$.

This form of CRN computation of a function, we call **stabilization**, enjoys a strong form of robustness since the computation can accomodate any perturbations on the variables during computation, and still compute the right result $O(I)$ once the input I is stabilized and the other variables are no longer externally perturbed.

Question 5. (2 points)

Give a CRN to compute the same function $O(I(0))$ **off-line** but not on-line, i.e. without allowing any external perturbation on at least one of the auxiliary variables.

Hint: you can re-use the previous CRN and its stabilization property to create, and prove the correctness of, a non-stabilizing CRN requiring a precise initial concentration of at least one of the auxiliary variables to compute the same function.

Answer

Simply by adding a reaction I => J for transforming the input I in a new auxiliary variable J given as input of the previous CRN structure, we get that:

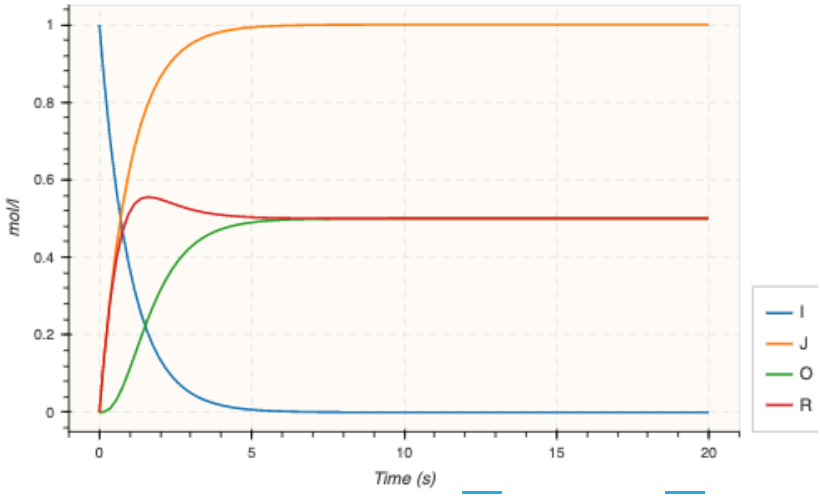
- *all $I(0)$ are transfered and added to $J(0)$*
- *the result is $O(I(0)) = (I(0) + J(0))/(1 + I(0) + J(0))$ according to the previous result*
- *and is thus sensitive to the initial value J(0) which should be precisely fixed to 0 in order to compute the same I/O function.*

In [10]:
```
clear_model.
I => J.
J+R => J+O.
_ => R.
R => _.
O => _.

present(I,i). present(R,r). present(O, o). present(J,j). parameter(i=1, r=1,

list_ode.
search_conservations.
```

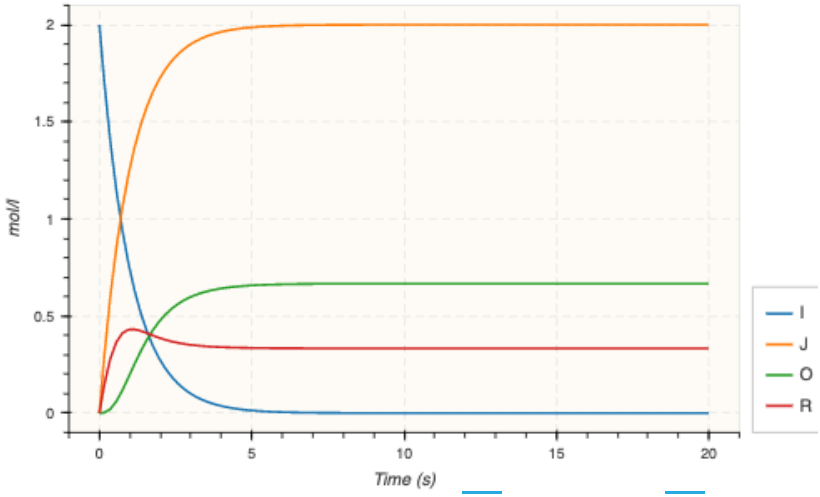
$$
\begin{aligned}
O_0 &= 0 \\
R_0 &= 1 \\
J_0 &= 0 \\
I_0 &= 1 \\
i &= 1 \\
r &= 1 \\
o &= 0 \\
j &= 0 \\
\frac{dO}{dt} &= J * R - O \\
\frac{dR}{dt} &= 1 - R - J * R \\
\frac{dJ}{dt} &= I \\
\frac{dI}{dt} &= -I
\end{aligned}
$$

Out[10]:
```
I+J
1 complex invariant(s)
```

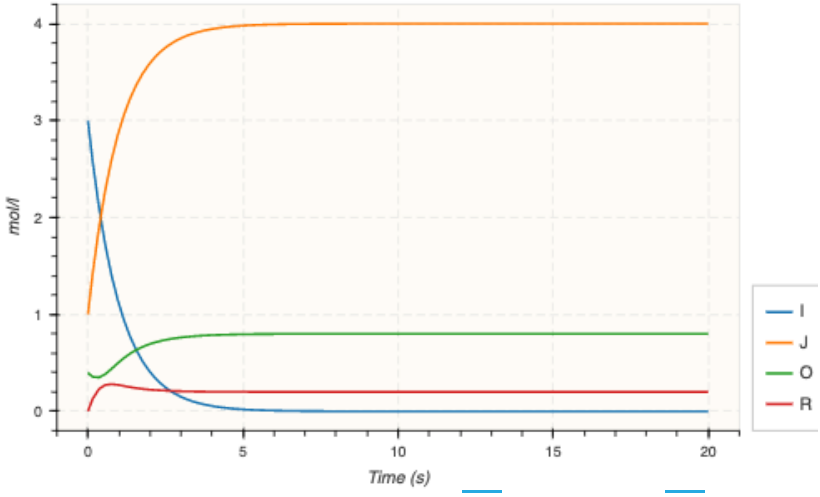
In [11]:
```
parameter(i=1, r=0, o=0, j=0). numerical_simulation. plot.
```



In [12]:
```
parameter(i=2, r=0, o=0, j=0). numerical_simulation. plot.
```



In [13]:
```
parameter(i=3, r=0, o=0.4, j=1). numerical_simulation. plot.
```



Question 6 (3 points)

Give a CRN to stabilize the norm function $n(x, y) = \sqrt{x^2 + y^2}$

Answer

In [14]:
```
stabilize_expression(n^2-x^2-y^2, n, [n = 0, x = 1, y=1]).
```

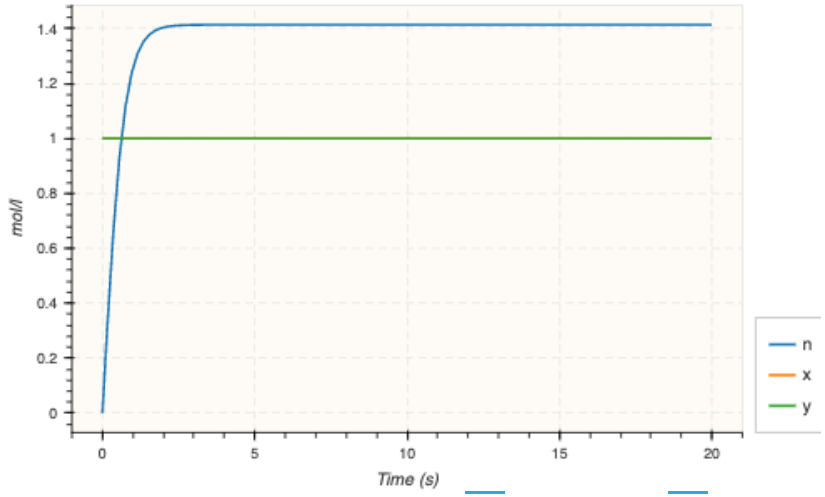
In [15]:
```
list_model.
```

Out[15]:
```
MA(1.0) for 2*x=>n+2*x.
MA(1.0) for 2*n=>n.
MA(1.0) for 2*y=>n+2*y.
initial_state(x=1).
initial_state(y=1).
parameter(
  i = 3,
  r = 0,
  o = 0.4,
  j = 1
).
```

In [16]:
```
list_ode.
```

$$
\begin{aligned}
n_0 &= 0 \\
x_0 &= 1 \\
y_0 &= 1 \\
i &= 3 \\
r &= 0 \\
o &= 0.4 \\
j &= 1 \\
\frac{dn}{dt} &= x^2 - n^2 + y^2 \\
\frac{dx}{dt} &= 0 \\
\frac{dy}{dt} &= 0
\end{aligned}
$$

In [17]:
```
numerical_simulation. plot.
```



In [18]:
```
validity_domain(F(G(n=v))).
```

Out[18]:
```
v=1.41421
```

Theory epilog

Mathieu Hemery showed last year that the set of real functions stabilized by a CRN with mass action law kinetics is precisely the set of algebraic functions, i.e. the set of real functions that are solutions of one polynomial equation, and are defined by one approximate solution point to disambiguate between the different parts of the solution curve, e.g. between the top and bottom function graphs of the circle defined by $x^2 + y^2 = 1$.

Question ∞ (bonus)

Any ideas to prove that result ?

Answer

See Mathieu Hemery, François Fages. Algebraic Biochemistry: a Framework for Analog Online Computation in Cells. In CMSB'22: Proceedings of the twentieth international conference on Computational Methods in Systems Biology, volume 13447 of Lecture Notes in BioInformatics. Springer-Verlag, 2022.

- preprint: https://hal.inria.fr/hal-03696273
- slides: https://lifeware.inria.fr/~fages/Papers/HF22slidesCMSB.pdf

Solution Written Examination MPRI 2.19 Biochemical Programming

Part II: on a CRN for estimating the derivative of an input signal

François Fages, March 2024

All questions are independent.

We consider a chemical reaction network (CRN) for estimating the derivative of an input signal x(t).

The CRN is defined by the following reaction rules with mass action law kinetics in BIOCHAM syntax.

The first reaction is just here to define real time and the input function of time $x(t)$, here $1 + sin(t)$.

In [1]:
```
initial_state(t=0).
MA(1) for _ => t.
function x = (1+sin(t)).
```

Now, the following reactions form the core derivative CRN with output function $d(t)$ defined as the difference of concentration between molecular species $d_p - d_m$.

The last reaction with fast kinetics is the standard annihilation reaction used in the dual-rail encoding of a variable d with possibly negative values as the difference of positive concentration between two molecular species $d_p - d_m$.

In [2]:
```
initial_state(xp=0, d_m=0, d_p=0).

MA(k*x) for _ =>  xp.
MA(k) for xp => _.
MA(k*l*x) for _ => d_p.
MA(k*l) for xp => xp + d_m.
MA(l) for d_p => _.
MA(l) for d_m => _.
MA(100) for d_p + d_m => _.

function d = d_p - d_m.
```

In [3]:
```
list_model.
```

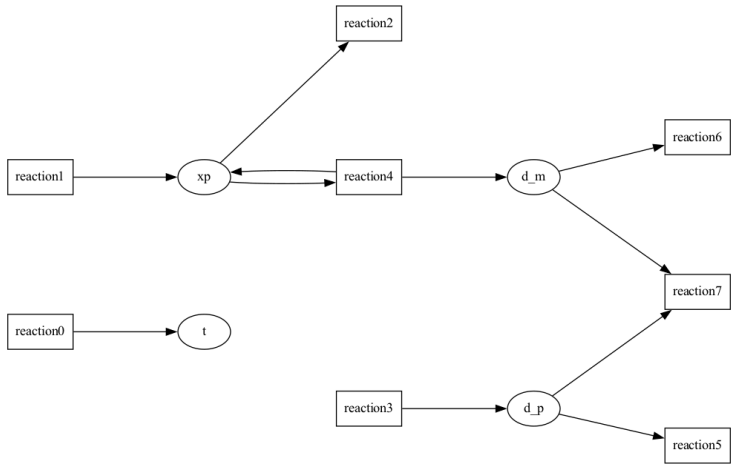
Out[3]:
```
MA(1) for _=>t.
MA(k*x) for _=>xp.
MA(k) for xp=>_.
MA(k*l*x) for _=>d_p.
MA(k*l) for xp=>d_m+xp.
MA(l) for d_p=>_.
MA(l) for d_m=>_.
MA(100) for d_m+d_p=>_.
absent(t).
absent(xp).
absent(d_m).
absent(d_p).
function(
  x = 1+sin(t),
  d = d_p-d_m
).
```

Question 1 (2 points)

Draw the species-reaction bipartite graph of that CRN

In [4]:
```
draw_reactions.
```

Out[4]:

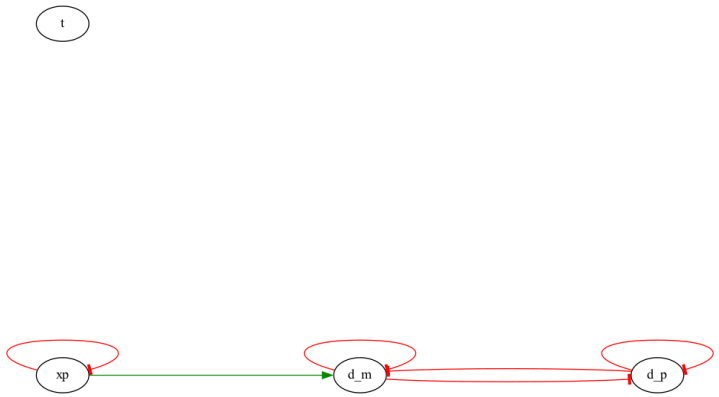


Question 2 (2 points)

Draw the graph of positive and negative influences between species in that CRN

Include the self influences on the species.

In [5]:
```
draw_influences.
```



Out[5]:

Question 3 ($\geq$ 2 points)

Define the symbolic transition Boolean constraint between the state variables of the 3 species and the 3 successor state variables

Note that the variable t used to define the input function is ignored here.

Hint: some of you might prefer to answer question 4 before.

Answer

The transition constraint on variables $(xp, d_m, d_p, xp', d'_m, d'_p)$ can be defined by the following formula in disjunctive normal form:

$(\neg xp \wedge xp' \wedge d'_m = d_m \wedge d'_p = d_p)$

$\vee(xp \wedge \neg xp' \wedge d'_m = d_m \wedge d'_p = d_p)$

$\vee(\neg d_p \wedge xp' = xp \wedge d'_m = d_m \wedge d'_p)$

$\vee(xp \wedge \neg d_m \wedge xp' = xp \wedge d'_m \wedge d'_p = d_p)$

$\vee(d_p \wedge xp' = xp \wedge d'_m = d_m \wedge \neg d'_p)$

$\vee(d_m \wedge xp' = xp \wedge \neg d'_m \wedge d'_p = d_p)$

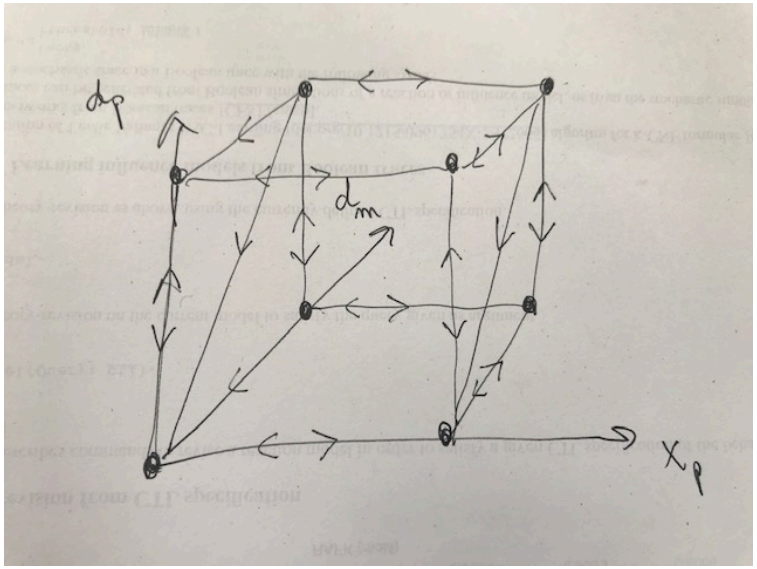
$\vee(d_m \wedge d_p \wedge xp' = xp)$

Question 4 ($\geq$ 2 points)

Draw the ground Boolean state transition graph between the 3 species represented by the vertices of a cube in the space (xp, dp, dm)

Hint: do not forget the transitions associated to the fast reaction

Answer



Question 5 ($\geq$ 2 points)

Give the variables (A, B) that satisfy the CTL formula not(not A Until B), i.e. checkpoint (A, B), from the initial state (0,0,0)

In [6]:
```
generate_ctl(checkpoint(A,B)).
```

Out[6]:
```
checkpoint(xp,d_m)
```

Question 6 ($\geq$ 2 points)

Give the Ordinary Differential Equations of the differential semantics of that CRN

In [7]:
```
list_ode.
```

$$
\begin{aligned}
d\_p_0 &= 0 \\
d\_m_0 &= 0 \\
xp_0 &= 0 \\
t_0 &= 0 \\
x &= 1 + sin(t) \\
d &= d\_p - d\_m \\
\frac{dd\_p}{dt} &= k * l * (1 + sin(t)) - 100 * d\_m * d\_p - d\_p * l \\
\frac{dd\_m}{dt} &= k * l * xp - 100 * d\_m * d\_p - d\_m * l \\
\frac{dxp}{dt} &= k * (1 + sin(t)) - k * xp \\
\frac{dt}{dt} &= 1
\end{aligned}
$$

Out[7]:

That CRN gives in output $d(t)$ an approximation of the derivative signal given by function $x(t)$.

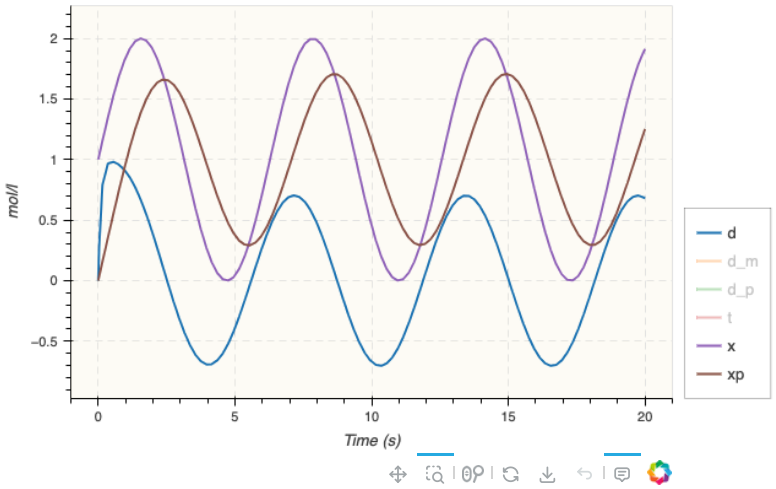
Here with $x(t) = 1 + sin(t)$, we expect $d(t) \sim cos(t)$.

By varying the value of parameter k we obtain the following simulation figures:

In [8]:
```
option(show:{xp, x, d}).
```

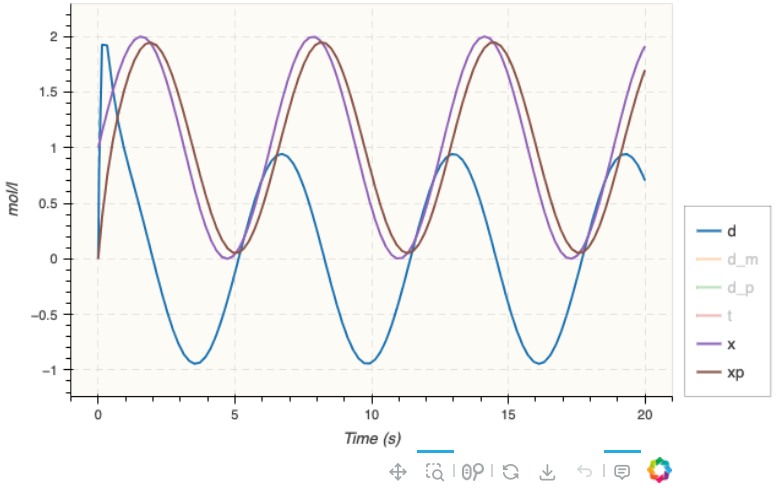
Out[8]:

In [9]:
```
parameter(l=10, k = 1).
numerical_simulation.
plot.
```



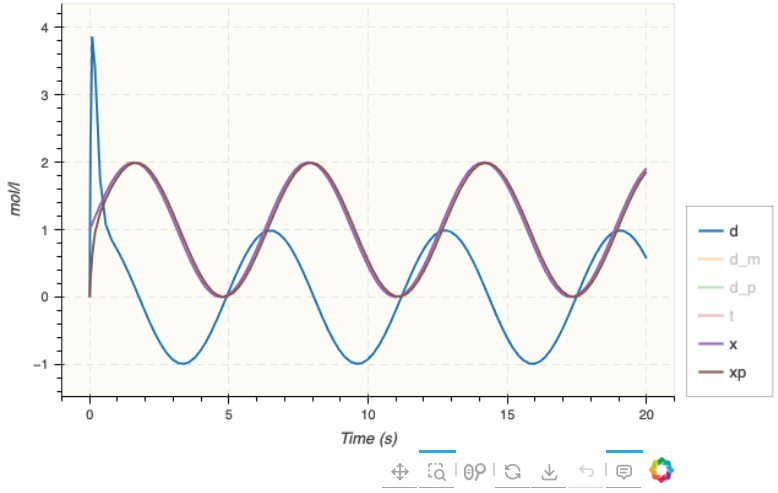
Out[9]:

In [10]:
```
parameter(l=10, k = 3).
numerical_simulation.
plot.
```



Out[10]:

In [11]:
```
parameter(l=10, k = 10).
numerical_simulation.
plot.
```



Out[11]:

Question 7 ($\geq$ 2 points)

Give the differential equation for the output function d(t)

Answer

$dd/dt = dd_p/dt - dd_m/dt = k.l.(x - xp) - l.(d_p - d_m)$

$= l.(((x - xp)/(1/k)) - d)$

Question 8 ($\geq$ 2 points)

Give an intuitive explanation why increasing the value of parameter k improves the approximation d of the time derivative function of the input function x(t)

Answer

One can think of $e = 1/k$ as the time delay separating $xp(t) \sim x(t - e)$.

The differential equation for $d(t)$ then expresses a finite difference method for estimating the derivative of $x(t)$.

The estimation is better for small values of e, that is when k increases.

Question 9 ($\geq$ 2 points)

Write a conjecture on the limit of d(t) as a function of x(t) when k tends to the infinity

Answer

In the view above, we have

$\lim_{k\to\infty} d(t) = dx/dt.$

Question 10 ($\geq$ 2 points)

Prove your conjecture

Answer

See

Hemery, Mathieu, Fages, François. On Estimating Derivatives of Input Signals in Biochemistry. In CMSB 2023 - 21st International Conference on Computational Methods in Systems Biology, volume 14137 of Lecture Notes in Computer Science. Springer-Verlag, 2023. [[https://inria.hal.science/hal-04154923]]