

Stochastic π -calculus modelling of gene regulatory networks

Ph.D. supervised by Olivier Roux and Morgan Magnin

Loïc Paulevé

Moves team, IRCCyN, Nantes, FR

January 15, 2009

Context

- Gene regulatory networks (**GRN**) modelling.
- Reproduce/understand particular behaviors.
- **Discrete** and **stochastic (temporal)** framework.

At which frequency our system will end up to stable behavior A?

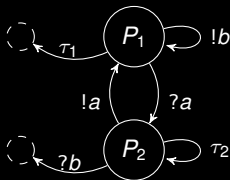
- **Process algebra** approach.

π -calculus modelling

- Concurrent processes algebra [Milner 89]
- Processes communicate using channels.
- P_1 calls (?) on channel a , P_2 answers (!) on channel a .
- Operators : parallelization, replication, name passing, name restriction.
- Turing complete.
- Here we will focus on π -calculus programs having a restricted grammar and operators.

Stochastic π -calculus modelling

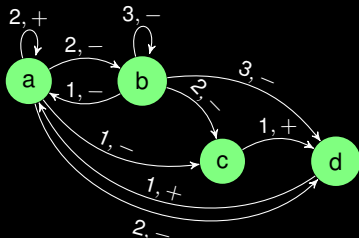
- Introduction of **delays** (τ) and **channels rates** [Priami 95]
- For each channel and delay we may specify an **use rate**.
- Rates control **probabilities of reactions** in the system (through the choice operator $+$).
- Rates control **average duration of reactions**.



$$\begin{aligned} P_1 &:= ?a.P_2 + \\ &\quad !b.P_1 + \\ &\quad \tau_1 \quad (P_1|P_2) \\ P_2 &:= !a.P_1 + \\ &\quad ?b + \\ &\quad \tau_2.P_2 \end{aligned}$$

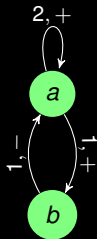
- Simulators : SPiM [Phillips & Cardelli 07], etc.

Generalized GRN dynamics



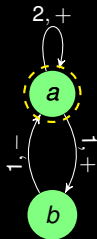
- Gene regulatory networks **graph** using **thresholds**.
- **Asynchronous** transitions: one gene at once changes level.
- A gene may see its level increase iff **at least one activator** is present.
- A gene may see its level decrease iff **at least one inhibitor** is present.

π -calculus modelling of GRN



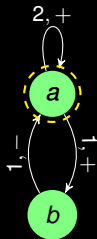
- One process per gene and per level.
- Answers to the call from its activators/inhibitors and goes to corresponding following/preceding process.

π -calculus modelling of GRN



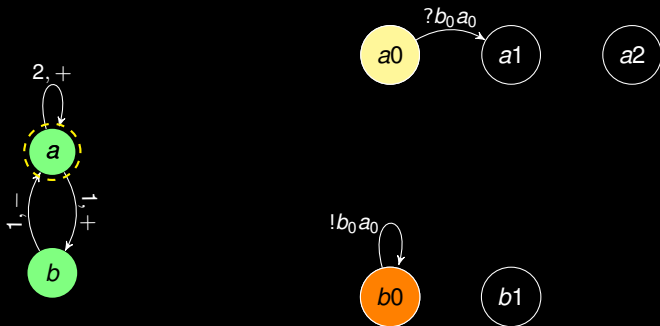
- One process per gene and per level.
- Answers to the call from its activators/inhibitors and goes to corresponding following/preceding process.

π -calculus modelling of GRN



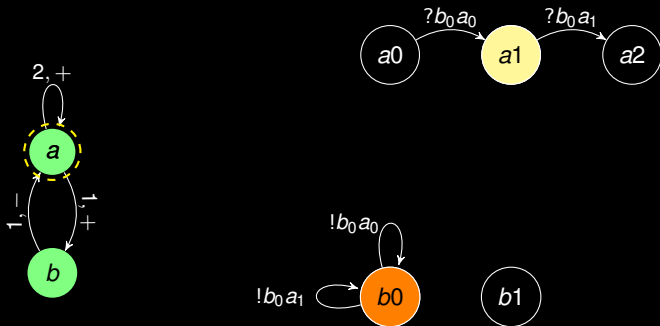
- One process per gene and per level.
- Answers to the call from its activators/inhibitors and goes to corresponding following/preceding process.

π -calculus modelling of GRN



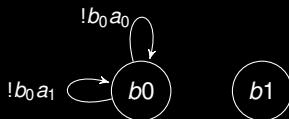
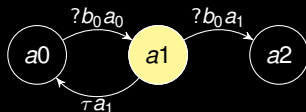
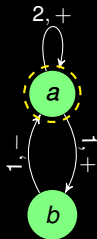
- One process per gene and per level.
- Answers to the call from its activators/inhibitors and goes to corresponding following/preceding process.

π -calculus modelling of GRN



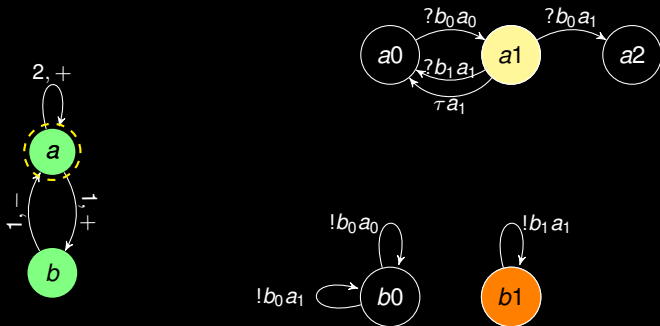
- One process per gene and per level.
- Answers to the call from its activators/inhibitors and goes to corresponding following/preceding process.

π -calculus modelling of GRN



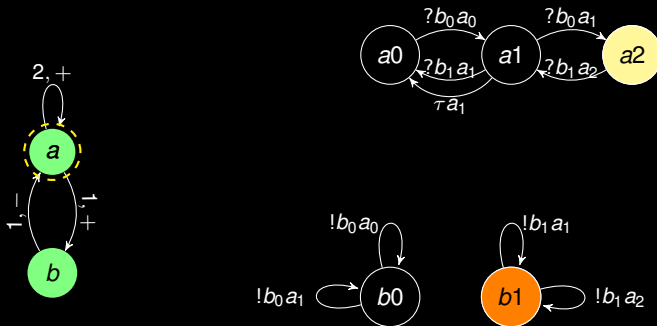
- One process per gene and per level.
- Answers to the call from its activators/inhibitors and goes to corresponding following/preceding process.

π -calculus modelling of GRN



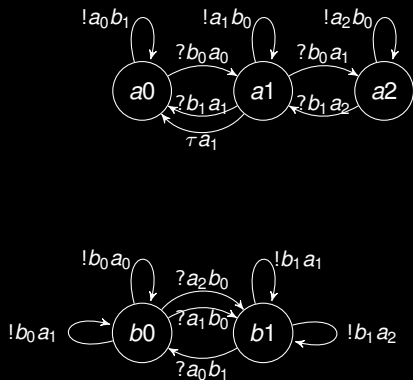
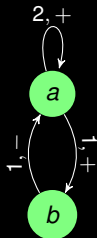
- One process per gene and per level.
- Answers to the call from its activators/inhibitors and goes to corresponding following/preceding process.

π -calculus modelling of GRN



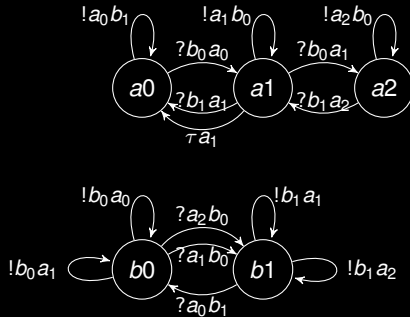
- One process per gene and per level.
- Answers to the call from its activators/inhibitors and goes to corresponding following/preceding process.

π -calculus modelling of GRN



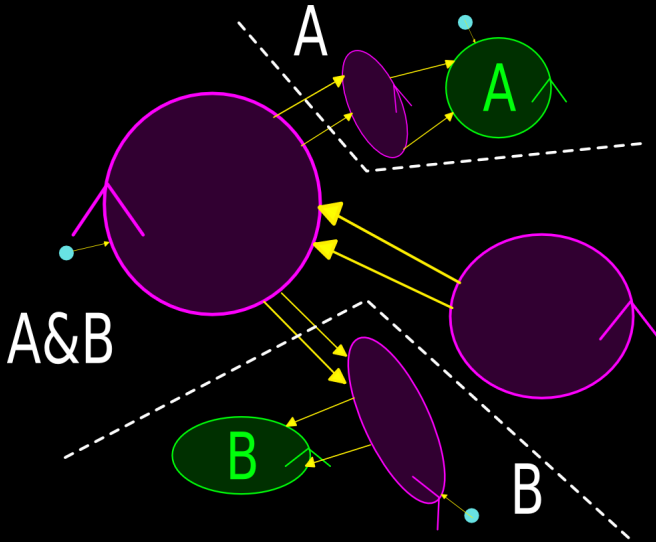
- One process per gene and per level.
- Answers to the call from its activators/inhibitors and goes to corresponding following/preceding process.

What do we look for?



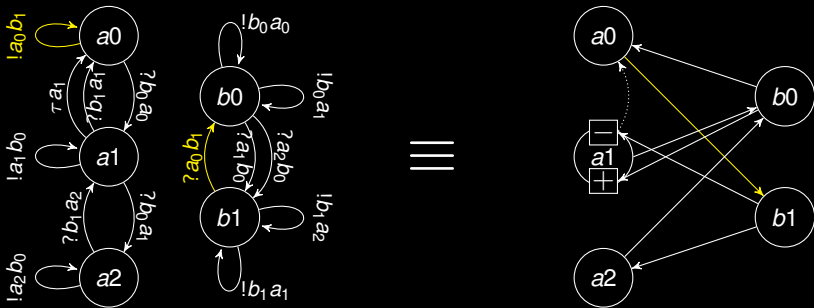
- Remark : our models have a very particular shape.
- Particular **patterns** inside our π -calculus models responsible of **particular parts of the dynamics**.
- How **assemble** these patterns to build a system respecting a given behavior?

Structures of dynamics

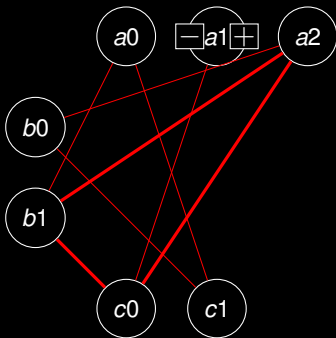
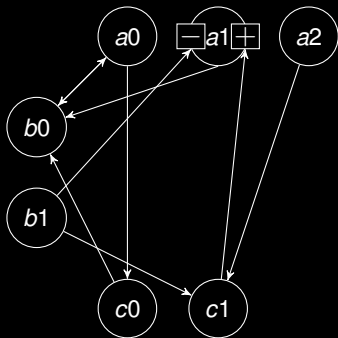


π -calculus reaction graphs

Well-suited representation of our π -calculus models :

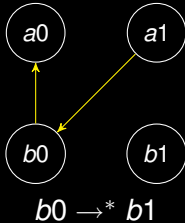


Pattern for stable states

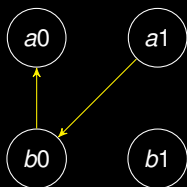


- n -clique in the complementary reaction graph
“Processes not sharing any channels”.
- Sufficient and necessary.

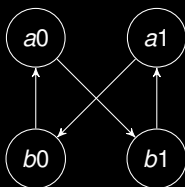
Constructing some particular oscillations



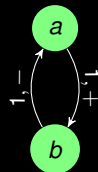
Constructing some particular oscillations



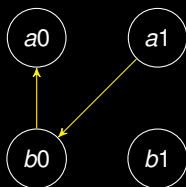
$b_0 \rightarrow^* b_1$



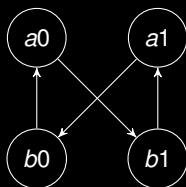
$b_0 \rightarrow^* b_1, b_1 \rightarrow^* b_0,$
 $a_0 \rightarrow^* a_1, a_1 \rightarrow^* a_0$



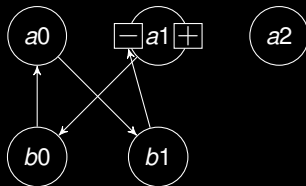
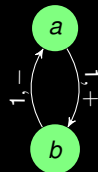
Constructing some particular oscillations



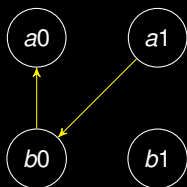
$b_0 \rightarrow^* b_1$



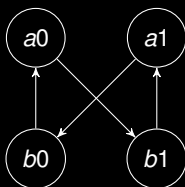
$b_0 \rightarrow^* b_1, b_1 \rightarrow^* b_0,$
 $a_0 \rightarrow^* a_1, a_1 \rightarrow^* a_0$



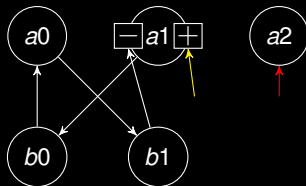
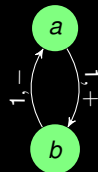
Constructing some particular oscillations



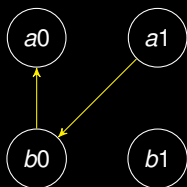
$b_0 \rightarrow^* b_1$



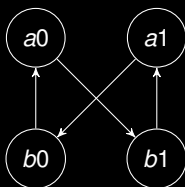
$b_0 \rightarrow^* b_1, b_1 \rightarrow^* b_0,$
 $a_0 \rightarrow^* a_1, a_1 \rightarrow^* a_0$



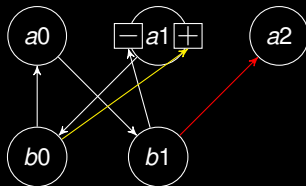
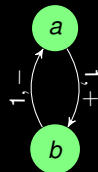
Constructing some particular oscillations



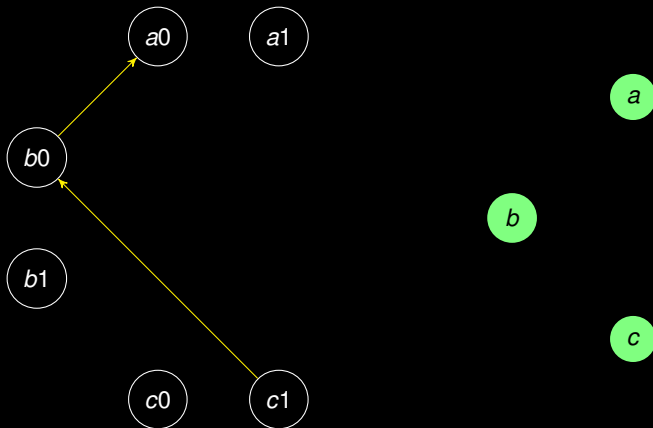
$b_0 \rightarrow^* b_1$



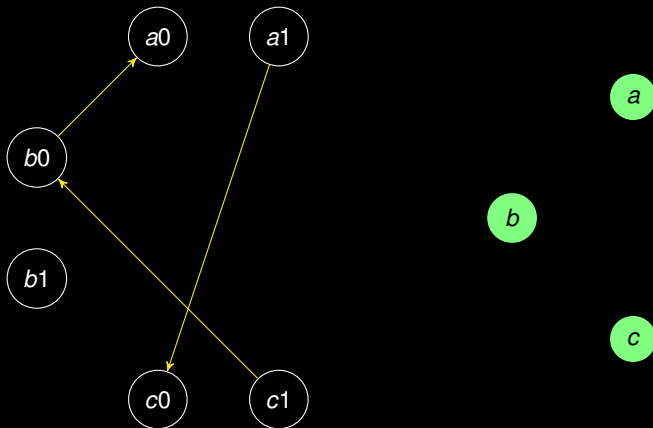
$b_0 \rightarrow^* b_1, b_1 \rightarrow^* b_0,$
 $a_0 \rightarrow^* a_1, a_1 \rightarrow^* a_0$



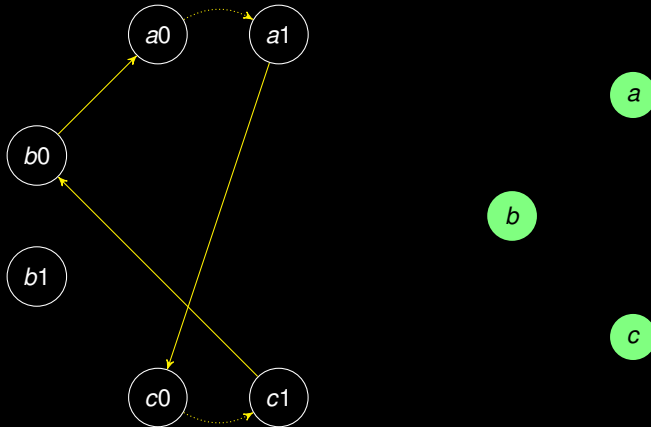
Constructing some particular oscillations



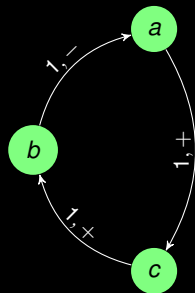
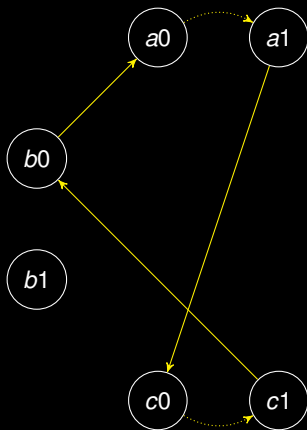
Constructing some particular oscillations



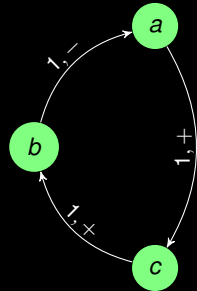
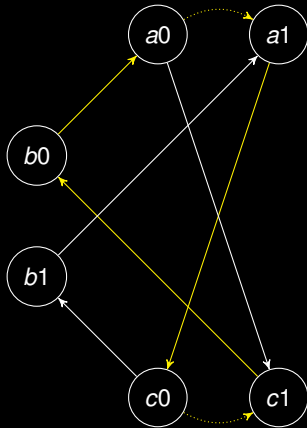
Constructing some particular oscillations



Constructing some particular oscillations



Constructing some particular oscillations



Conclusion

- We showed how modelling **generalized GRN** dynamics in π -calculus,
- this gave us models with a particular shape.
- We defined a well-suited representation of these models.
- We showed some **particular patterns** leading to the presence of **particular behaviors**.
- We showed how to **build** systems respecting some structures of wanted dynamics.

Ongoing works and perspectives

- Provide a framework to easily **program wanted behaviors**.
- Infer from patterns some important channels
⇒ stochastic **parameters inference**.
- Enlarge to deterministic GRN dynamics (Thomas' K)
(shape of models is changing, but not so much).
- π -calculus patterns to GRN graph.

Questions ?

Thank you for your attention !