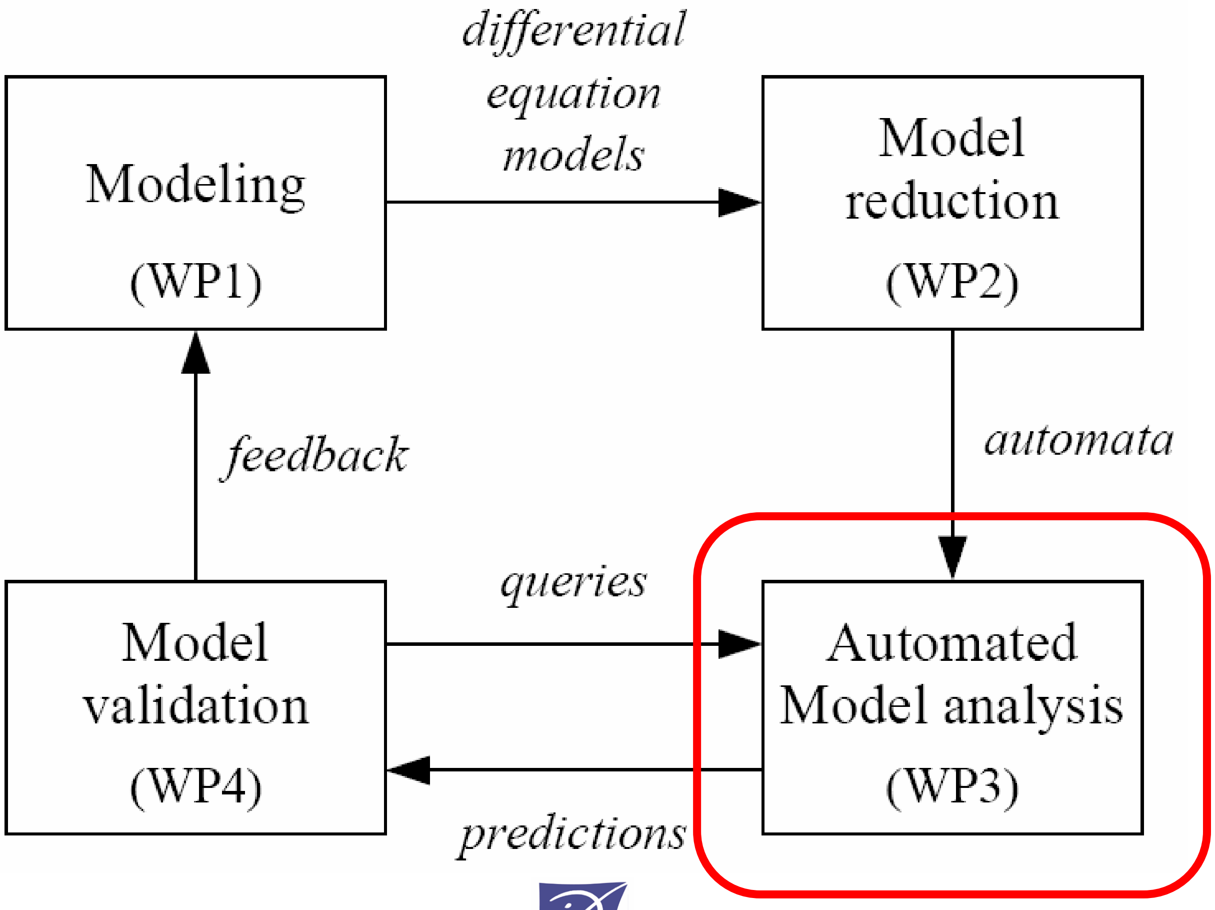
Computation Tree Regular Logic for Genetic Regulatory Networks

Radu Mateescu, Pedro Tiago Monteiro,
Estelle Dumas, and Hidde de Jong

INRIA Grenoble - Rhône-Alpes
VASY and IBIS project-teams

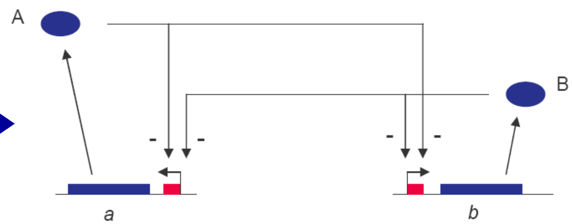
Context

- EC-MOAN (*Escherichia Coli - MOdeling and ANalysis*) European project FP6-NEST-PATH-COM no. 043235



Analysis of genetic regulatory networks
(GNA – Genetic Network Analyzer)

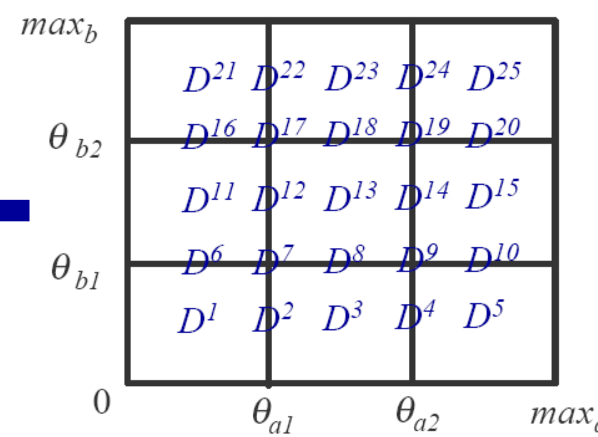
E. coli



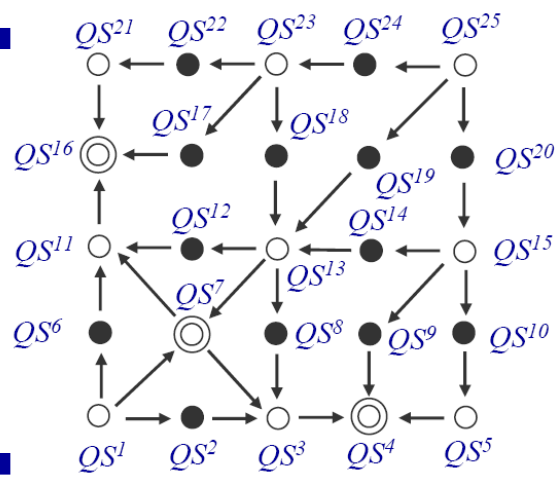
Genetic Regulatory Network

$$\dot{x}_a = \kappa_a\, s^-(x_a\,,\, \theta_{a2})\; s^-(x_b\,,\, \theta_{b1}\,) - \gamma_a\, x_a$$
$$\dot{x}_b = \kappa_b\, s^-(x_a\,,\, \theta_{a1})\; s^-(x_b\,,\, \theta_{b2}\,) - \gamma_b\, x_b$$

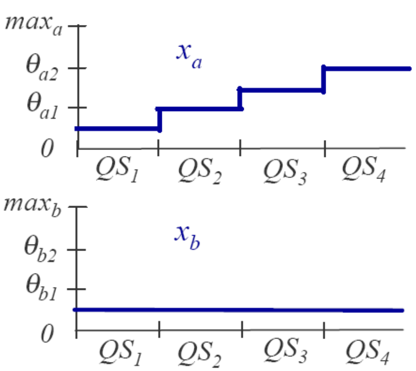
Piecewise-linear differential equations



Qualitative simulation



State-transition graph

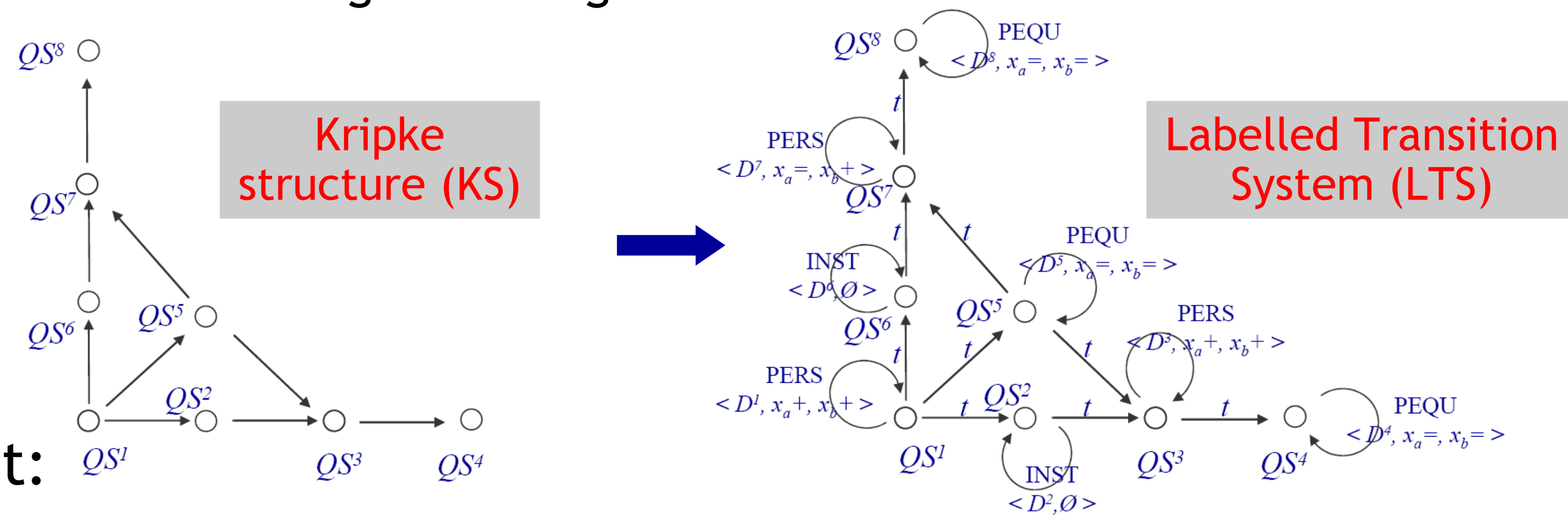


Simulation (GNA)

Model checking (CADP)

A first connection from GNA to CADP
[Batt-Bergamini-deJong-Garavel-Mateescu-04]

- GNA2BCG: translation from KSs to LTSs
  - Succinct (same number of states)
  - Reduction using branching bisimulation



- But:
  - Does not preserve strong bisimulation (extra self-loops)
  - Properties expressed on LTSs (action-based logics)
  - Difficult to relate with the input GNA model
  - Requires expertise in model checking

Motivation of current work

Devise a temporal logic that:

- Is state-based (interpreted directly on KSs)
- Is powerful enough to capture biological properties
  - Multistability (branching-time)
  - Oscillations (linear-time)
- Has a reasonable model checking complexity
  - Preferably linear-time w.r.t. the KS size
- Has a user-friendly syntax for non-experts
  - Succinct and intuitive formulation of properties
  - A small number of temporal operators

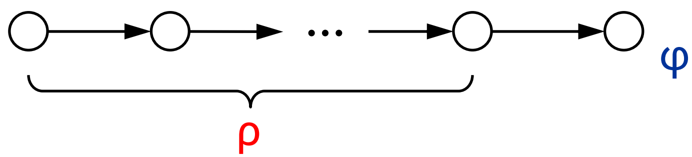
Computation Tree Regular Logic — CTRL
(syntax)

$\varphi ::= p$	*atomic proposition*	state formulas
$\mid \neg\varphi \mid \varphi_1 \vee \varphi_2$	*boolean connectors*	
$\mid EF_\rho\, \varphi$	*potentiality*	
$\mid AF_\rho\, \varphi$	*inevitability*	
$\mid EF^\infty_\rho$	*potential looping*	
$\mid AF^\infty_\rho$	*inevitable looping*	

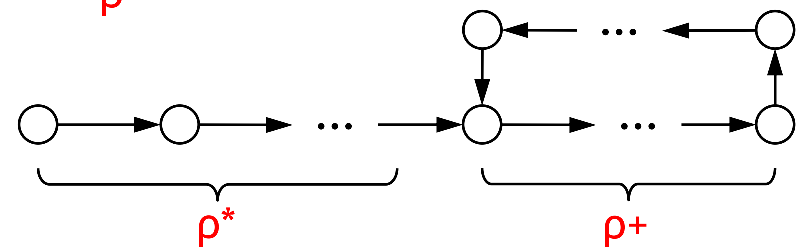
$\rho ::= \varphi$	*one-step interval*	regular formulas
$\mid \rho_1 . \rho_2$	*concatenation*	
$\mid \rho_1 \mid \rho_2$	*choice*	
$\mid \rho^*$	*iteration 0 or more times*	

CTRL — state formulas
(semantics)

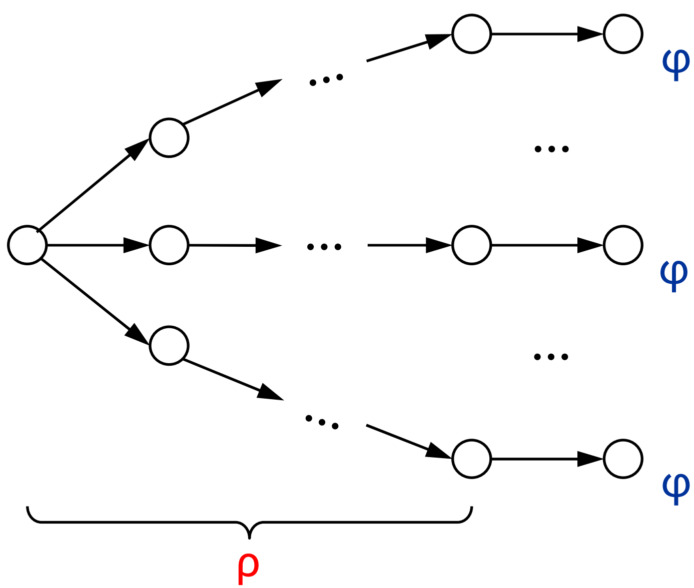
- $EF_{\rho}\ \varphi$



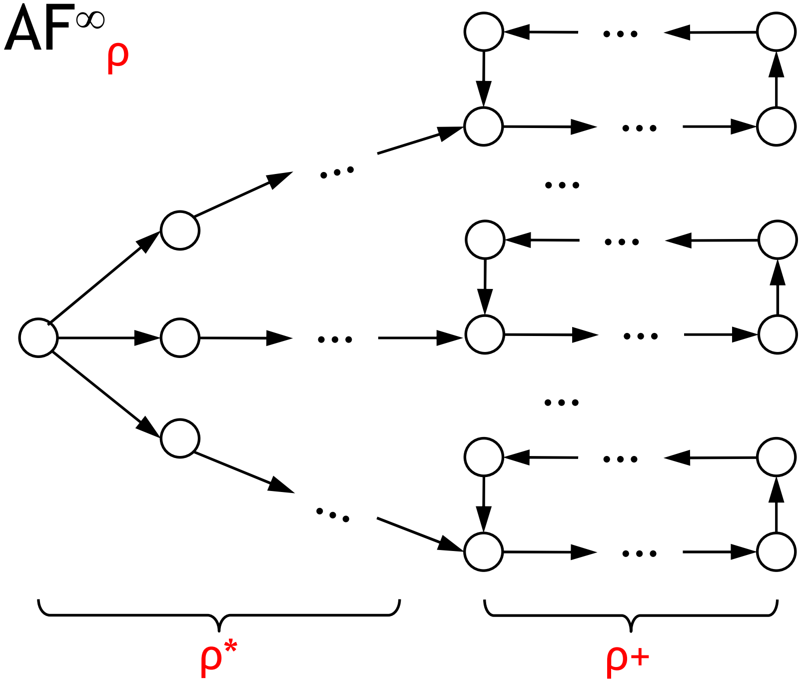
- EF^{∞}_{ρ}



- $AF_{\rho}\ \varphi$

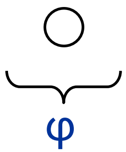


- AF^{∞}_{ρ}

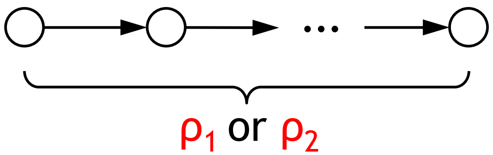


CTRL — regular formulas
(semantics)

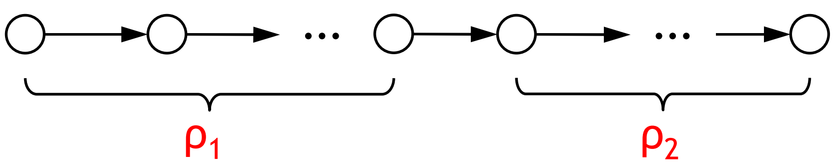
- φ



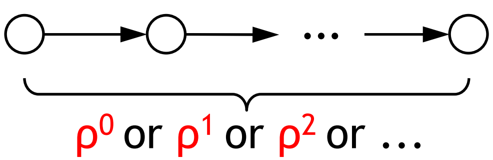
- $\rho_1 \mid \rho_2$



- $\rho_1 \cdot \rho_2$



- ρ^*



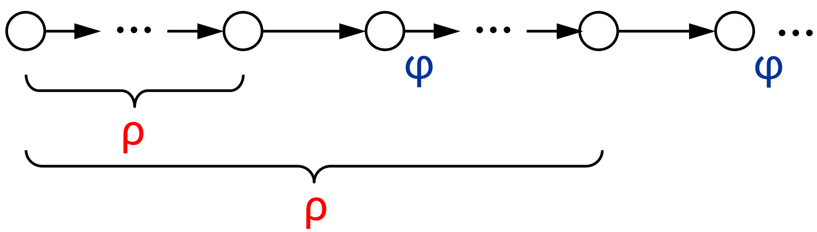
CTRL – derived operators
(syntax)

$EG_\rho\ \varphi = \neg AF_\rho\ \neg\varphi$	*trajectory*	state formulas
$AG_\rho\ \varphi = \neg EF_\rho\ \neg\varphi$	*invariance*	
$EG^{\perp}_{\rho} = \neg AF^{\infty}_{\rho}$	*potential saturation*	
$AG^{\perp}_{\rho} = \neg EF^{\infty}_{\rho}$	*inevitable saturation*	

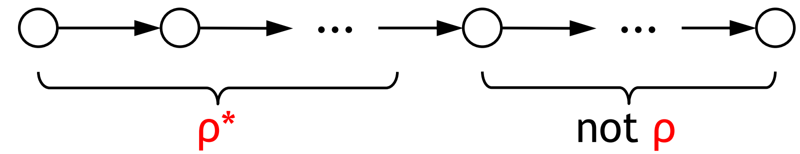
		regular formulas
nil = false*	*empty interval*	
$\rho+ = \rho\ .\ \rho^*$	*iteration 1 or more times*	

CTRL — derived operators
(semantics)

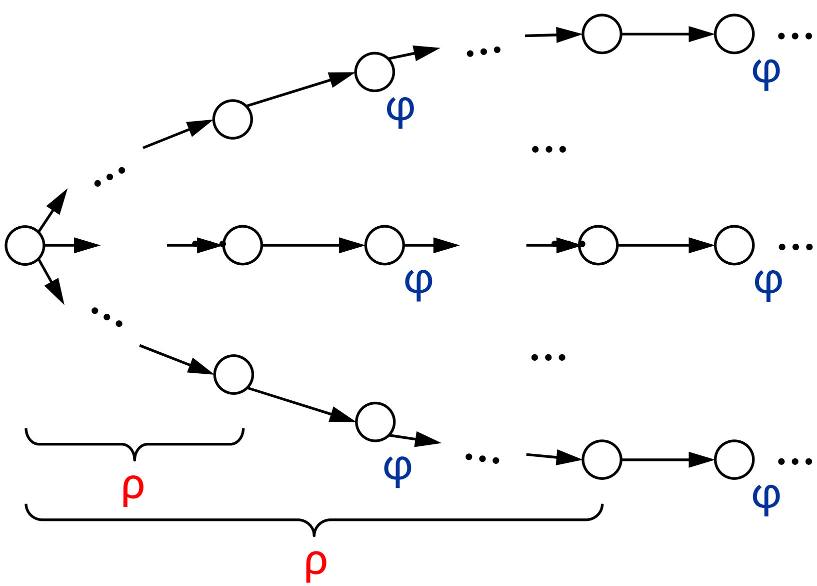
- $EG_{\rho}\ \varphi$



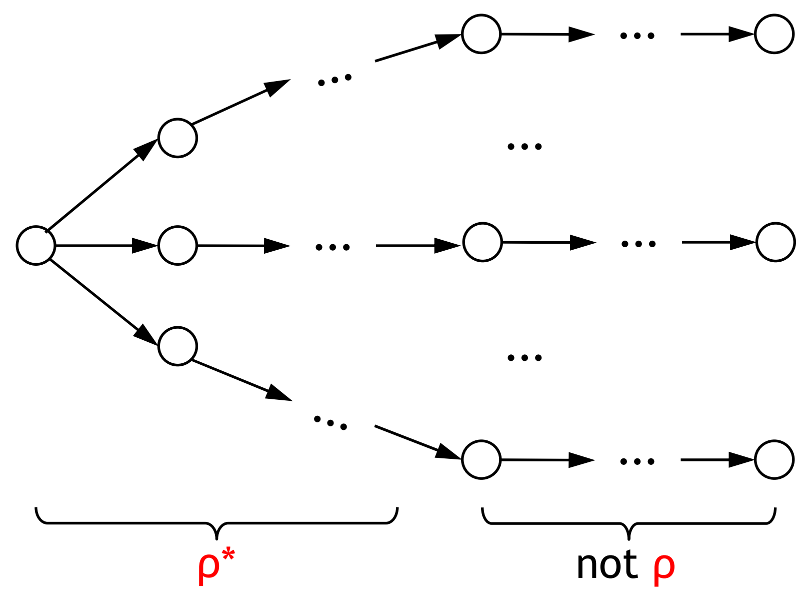
- $EG^{\perp}_{\rho}$



- $AG_{\rho}\ \varphi$



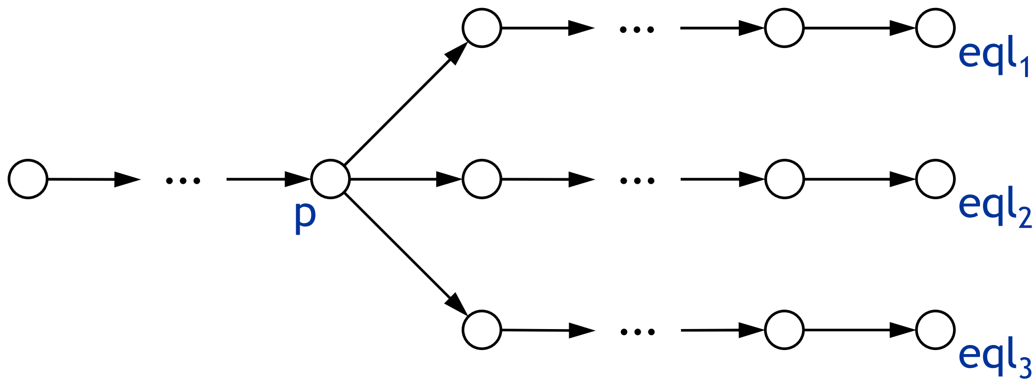
- $AG^{\perp}_{\rho}$



Examples
(biological properties)

- **Multistability:**

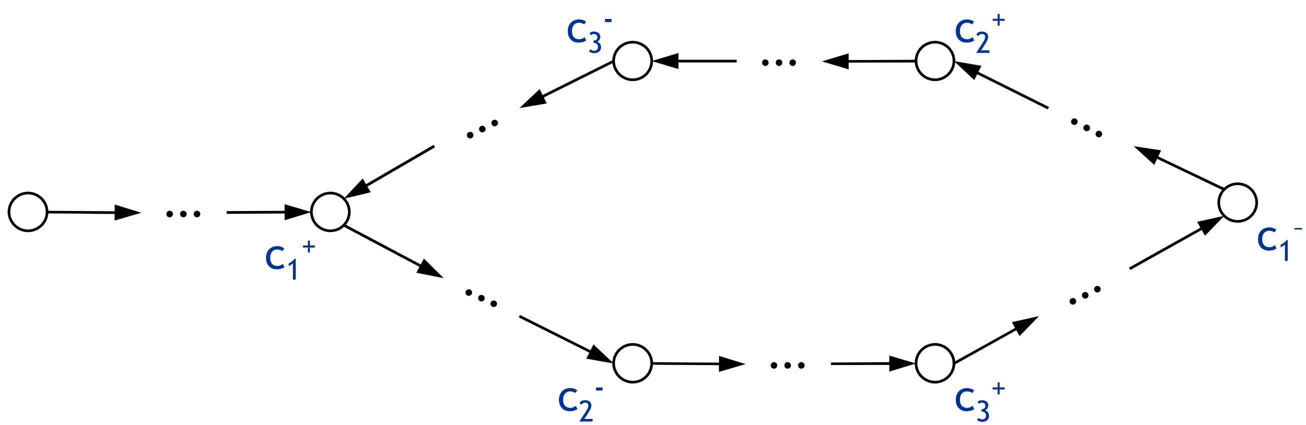
reachability of several equilibrium states



$AG_{true^*}\ (p \Rightarrow (EF_{true^*}\ eql_1 \wedge EF_{true^*}\ eql_2 \wedge EF_{true^*}\ eql_3))$

- **Oscillations:**

existence of complex cycles



$EF_{true^*}\ EF^{\infty}_{c1+\ .\ true^*.\ c2-\ .\ true^*.\ c3+\ .\ true^*.\ c1-\ .\ true^*.\ c2+\ .\ true^*.\ c3-\ .\ true^*}$

Examples
(concurrent system properties)

- Alternation between send/receive (safety):

$$AG_{(nil \mid (true^*.rcv)).(\neg snd)^*.rcv) \mid (true^*.snd.(\neg rcv)^*.snd)}\ false$$

CTL formulation:

$\neg E\ [\neg snd\ U\ rcv] \wedge AG\ (rcv \Rightarrow \neg E\ [\neg snd\ U\ rcv]) \wedge AG\ (snd \Rightarrow \neg E\ [\neg rcv\ U\ snd])$

- Inevitable reception after possible errors (liveness):

$$AG_{true^*.snd}\ AF_{(true^*.err)^*.rcv}\ true$$

- Bounded overtaking (fairness):

$$AG_{true^*.req1}\ AG^{\perp}_{(\neg get1)^*.req2.(\neg get1)^*.get2}$$

LTL formulation:

$$G(req_1 \Rightarrow ((get_1\ R\ \neg req_2) \vee (\neg get_1\ U\ ((req_2 \wedge (get_1\ R\ \neg get_2)) \vee (get_2 \wedge (get_1\ R\ \neg req_2))))))$$

Expressiveness of CTRL

- CTRL subsumes CTL (Computation Tree Logic)

  $E [\varphi \, U \, \psi] = EF_{\varphi^*} \, \psi$

  $A [\varphi \, U \, \psi] = AF_{\varphi^*} \, \psi$

  the until operator U is not primitive in CTRL

- CTRL subsumes LTL (Linear Time Logic)

  $EF^{\infty}_{true^* \, . \, final \, . \, true}$

  acceptance condition in Büchi automata

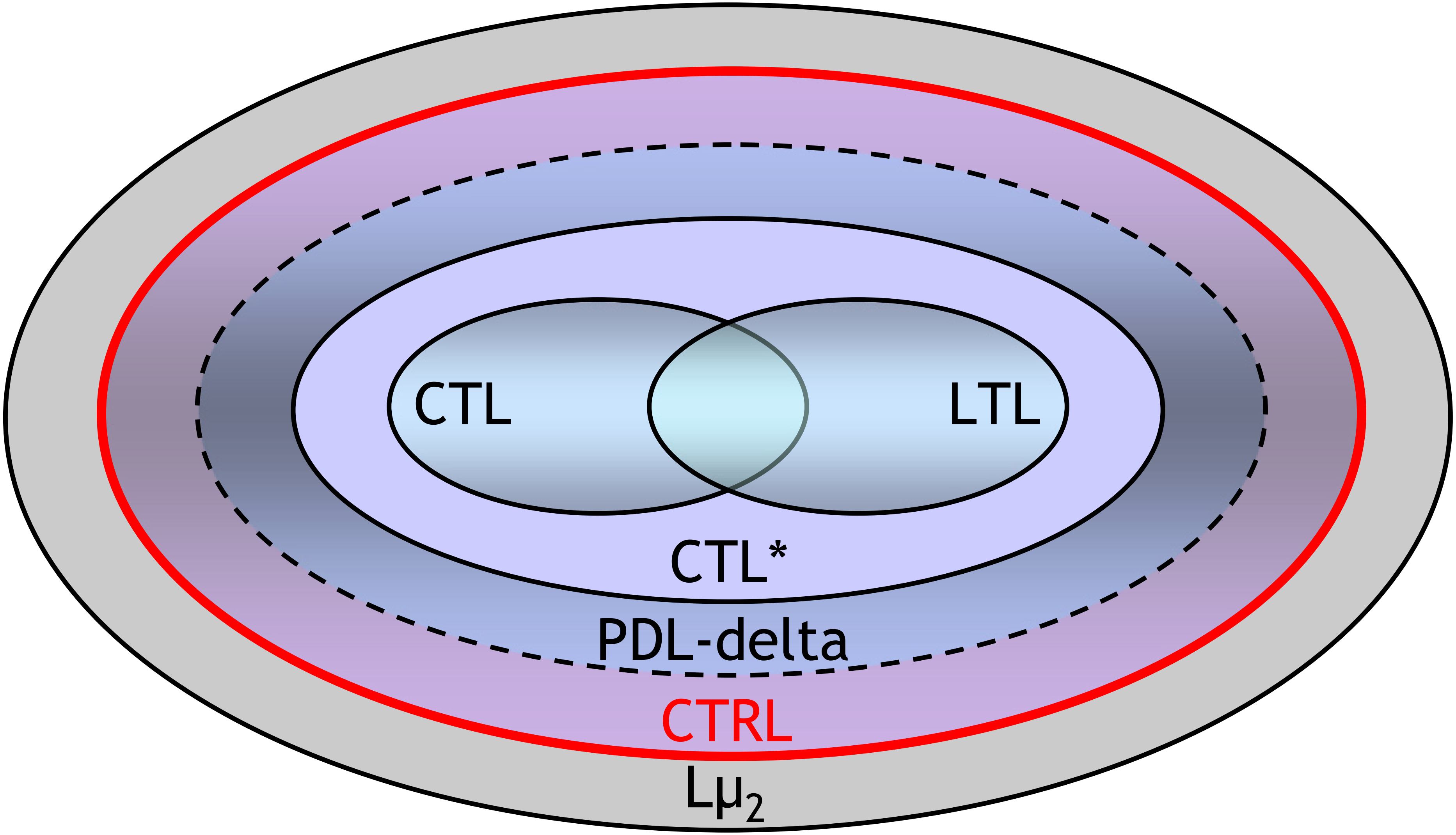
- CTRL subsumes CTL*

  $\{ p, \neg\varphi, \varphi_1 \vee \varphi_2, EF_{\rho} \, \varphi, EF^{\infty}_{\rho} \}$

  $\approx$ PDL-delta $\supseteq$ CTL*

  CTRL fragment corresponding to a state-based counterpart of PDL-delta

Expressiveness of CTRL

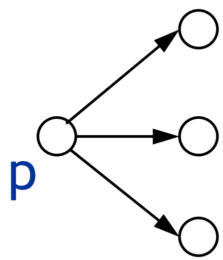


On-the-fly model checking approach

- Avoid building a CTRL model checker from scratch
  ➔ *reuse verification technology available in CADP*

<u>State-based world</u> | <u>Action-based world</u>

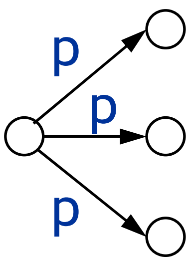
- KSs



Classical translation

- succinct (same |S| and |T|)
- preserves strong bisimulation
- can be done on-the-fly

- LTSs



- CTRL

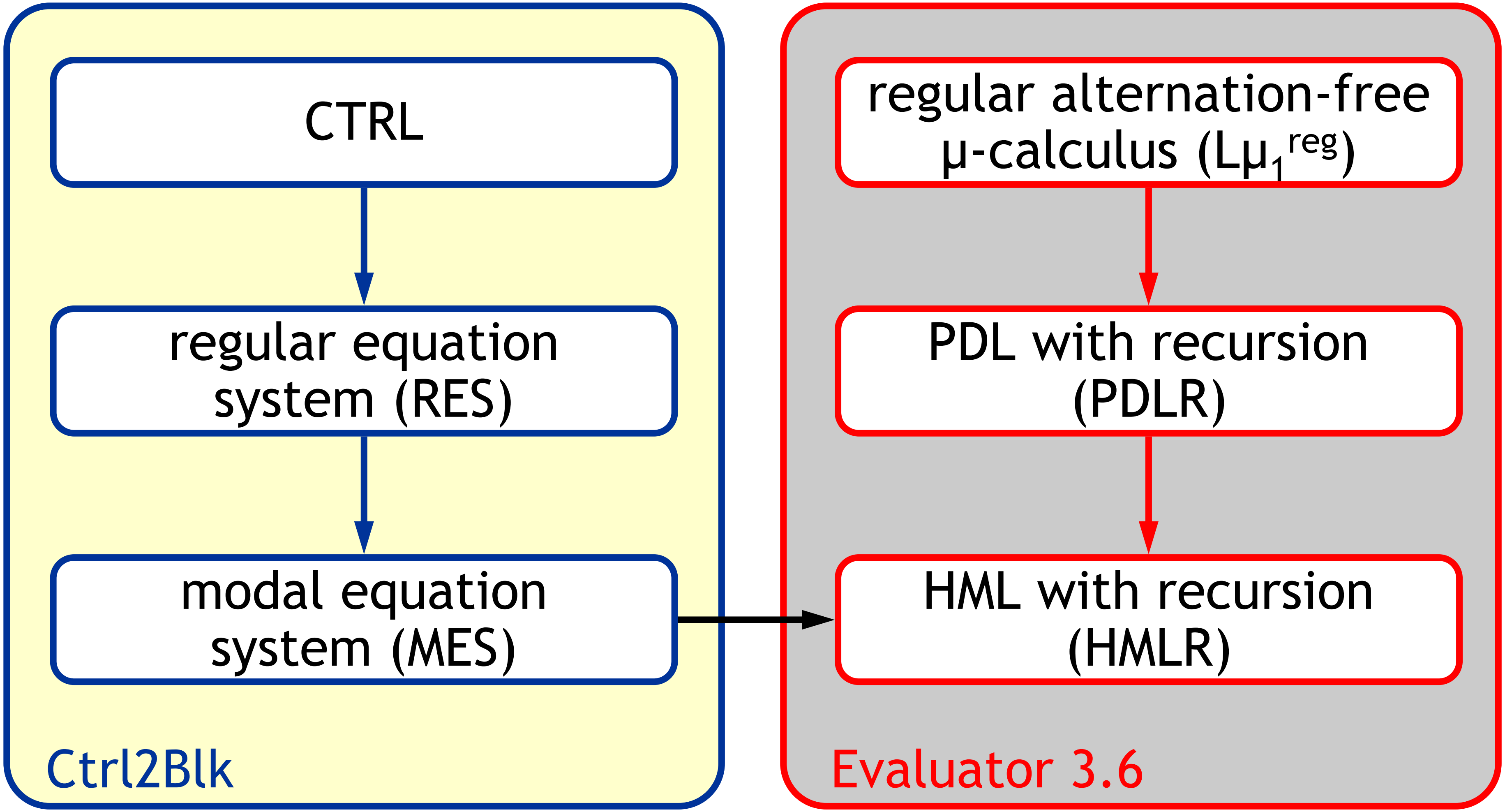
φ

New translation proposed

- Modal μ-calculus

?

Translation approach



Translation from CTRL to RES

CTRL

$AG_{(p \mid q)^*.r} \left(AF_{((p^*.q) \mid r^*)^*.q^*}\ r \vee EF^{\infty}_{true^*.p.true^*.q}\right)$

RES

$\{ X_1 =_\nu AG_{(p \mid q)^*.r}\, X_2 \ ,\ X_2 =_\nu Y_1 \vee Z_1 \}$

$\{ Y_1 =_\mu AF_{((p^*.q) \mid r^*)^*.q^*}\ r \}$

$\{ Z_1 =_\nu EF^{\infty}_{true^*.p.true^*.q}\ Z_1 \}$

Translation from RES to MES
(operators EF_ρ and AG_ρ)

Apply PDL-like identities:

- $AG_{\rho 1 \, . \, \rho 2} \; \varphi = AG_{\rho 1} \; AG_{\rho 2} \; \varphi$
- $AG_{\rho *} \; \varphi = \varphi \wedge AG_{\rho} \; AG_{\rho *} \; \varphi$
- $AG_{\rho 1 \mid \rho 2} \; \varphi = AG_{\rho 1} \; \varphi \wedge AG_{\rho 2} \; \varphi$

RES

$$\{ X_1 =_\nu AG_{(p \mid q)*.r} X_2 \; , \; X_2 =_\nu Y_1 \vee Z_1 \}$$

MES

$$\{ X_1 =_\nu X_3 \wedge X_4 \, , \, X_2 =_\nu Y_1 \vee Z_1 \, ,$$
$$X_3 =_\nu AG_r \, X_2 \; , \; X_4 =_\nu AG_p \, X_1 \wedge AG_q \, X_1 \}$$

Translation from RES to MES
(operators AF_ρ and EG_ρ)

- No PDL-like identities hold for $AF_\rho\ \varphi$

  ➔ *propose a different translation scheme*

Step 1: Translation to potentiality form (PF)

$\{ Y_1 =_\mu AF_{((p^*.q) \mid r^*)^*.q^*}\ r \}$ — A := E → $\{ Y_1 =_\mu EF_{((p^*.q) \mid r^*)^*.q^*}\ r \}$

↓ PDL-like identities

PF

$$\{ Y_1 =_\mu Y_2 \vee Y_3,\ Y_2 =_\mu Y_4 \vee Y_5,\ Y_3 =_\mu Y_6 \vee Y_7,\ Y_4 =_\mu r,$$
$$Y_5 =_\mu EF_q\ Y_2,\quad Y_6 =_\mu Y_8 \vee Y_9,\quad Y_7 =_\mu Y_1 \vee Y_{10},$$
$$Y_8 =_\mu EF_q\ Y_1,\quad Y_9 =_\mu EF_p\ Y_6,\quad Y_{10} =_\mu EF_r\ Y_7 \}$$

Translation from RES to MES
(operators AF_ρ and EG_ρ)

Step 2: Translation to guarded potentiality form (GPF)

PF

$$\{ Y_1 =_\mu Y_2 \vee Y_3,\ Y_2 =_\mu Y_4 \vee Y_5,\ Y_3 =_\mu Y_6 \vee Y_7,\ Y_4 =_\mu r,$$
$$Y_5 =_\mu EF_q\, Y_2,\ \ Y_6 =_\mu Y_8 \vee Y_9,\ \ Y_7 =_\mu Y_1 \vee Y_{10},$$
$$Y_8 =_\mu EF_q\, Y_1,\ \ Y_9 =_\mu EF_p\, Y_6,\ \ Y_{10} =_\mu EF_r\, Y_7 \}$$

absorption: $Y =_\mu Y \vee \varphi \ \equiv\ Y =_\mu \varphi$

GPF (≈ derivatives of regular expressions [Brzozowski-64])

$$\{ Y_1 =_\mu EF_p\, Y_3 \vee EF_q\, Y_1 \vee EF_q\, Y_2 \vee Y_4,$$
$$Y_2 =_\mu EF_q\, Y_2 \vee Y_4,\ Y_3 =_\mu EF_p\, Y_3 \vee EF_q\, Y_1,\ Y_4 =_\mu r \}$$

Translation from RES to MES
(operators AF_ρ and EG_ρ)

Step 3: Determinization

GPF

$$\{ Y_1 =_\mu EF_p\ Y_3 \vee EF_q\ Y_1 \vee EF_q\ Y_2 \vee Y_4\ ,$$
$$Y_2 =_\mu EF_q\ Y_2 \vee Y_4,\ Y_3 =_\mu EF_p\ Y_3 \vee EF_q\ Y_1,\ Y_4 =_\mu r \}$$

determinization:

$$Y_{\{1\}} =_\mu AF_p\ Y_{\{3\}} \vee AF_q\ Y_{\{1,2\}} \vee AF_{p \wedge q}\ Y_{\{1,2,3\}} \vee Y_4$$

simplification: $AF_{p \wedge q}\ \varphi \Rightarrow AF_q\ \varphi$

MES

$$\{ Y_{\{1\}} =_\mu AF_p\ Y_{\{3\}} \vee AF_q\ Y_{\{1\}} \vee Y_{\{4\}}\ ,$$
$$Y_{\{3\}} =_\mu AF_p\ Y_{\{3\}} \vee AF_q\ Y_{\{1\}},\ Y_{\{4\}} =_\mu r \}$$

Translation from RES to MES
(operators EF^{∞}_{ρ}, $AG^{\perp}_{\rho}$, AF^{∞}_{ρ} and $EG^{\perp}_{\rho}$)

similar procedure applies for AF^{∞}_{ρ}

RES

$$\{ Z_1 =_{\nu} EF^{\infty}_{true^*.p.true^*.q}\ Z_1 \}$$

↓

RES of alternation depth 2

$$\{ Z_0 =_{\nu} Z_1 \} . \{ Z_1 =_{\mu} EF^{\infty}_{true^*.p.true^*.q}\ Z_0 \}$$

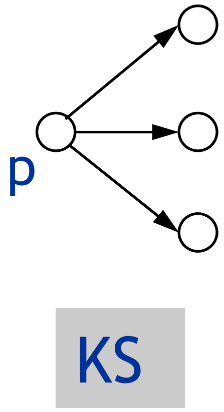
↓ PDL-like identities

MES of alternation depth 2

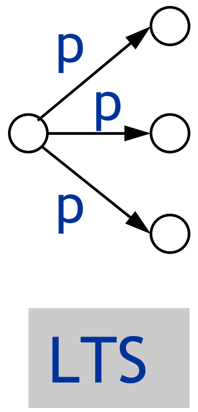
$$\{ Z_0 =_{\nu} Z_1 \} .$$
$$\{ Z_1 =_{\mu} EF_p\ Z_2 \vee EF_{true}\ Z_1,\ Z_2 =_{\mu} EF_q\ Z_0 \vee EF_{true}\ Z_2 \}$$

Translation from MES to HMLR

- Right-hand sides of MES equations:
  - Combinations of elementary CTRL modalities $EF_{\varphi}Y$, $AF_{\varphi}Y$, $EG_{\varphi}Y$, $AG_{\varphi}Y$
- Must convert CTRL modalities to HML modalities
  - Take into account the translation from KS to LTS



CTRL formula	HML formula
p	$\langle p \rangle \mathsf{true}$
$\mathsf{EF}_p X$	$\langle p \rangle X$
$\mathsf{AG}_p X$	$[p] X$
$\mathsf{AF}_p X$	$\langle p \rangle \mathsf{true} \wedge [\mathsf{true}] X$
$\mathsf{EG}_p X$	$\langle p \rangle \mathsf{true} \Rightarrow \langle \mathsf{true} \rangle X$



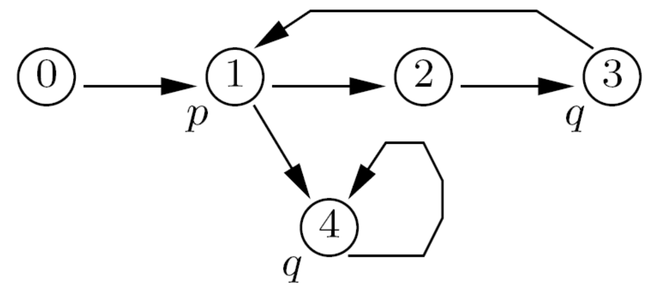
On-the-fly model checking

(illustration on $\mathsf{EF}^{\infty}{}_{\rho}$)

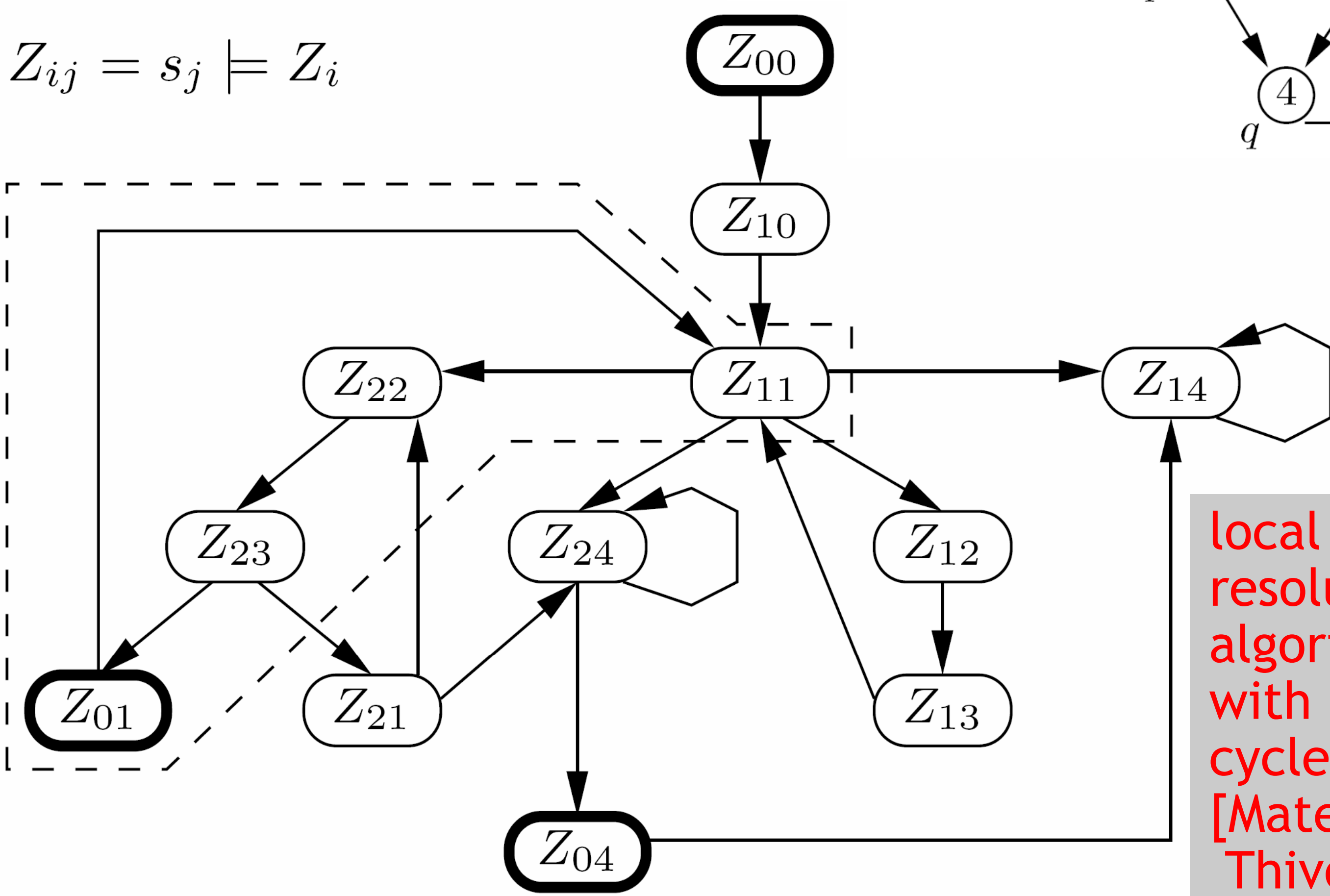
Formula: $\mathsf{EF}^{\infty}_{\mathsf{true}^*.p.\mathsf{true}^*.q}$

MES:
$$\left\{\begin{array}{l} Z_0 \stackrel{\mu}{=} Z_1 \\ Z_1 \stackrel{\mu}{=} \mathsf{EF}_p Z_2 \vee \mathsf{EF}_{\mathsf{true}} Z_1 \\ Z_2 \stackrel{\mu}{=} \mathsf{EF}_q Z_0 \vee \mathsf{EF}_{\mathsf{true}} Z_2 \end{array}\right\}$$

KS:



BES: $Z_{ij} = s_j \models Z_i$



abusive merge of the 2 equation blocks into one μ-block

local BES resolution algorithm $A4_{cyc}$ with marked cycle detection [Mateescu-Thivolle-08]

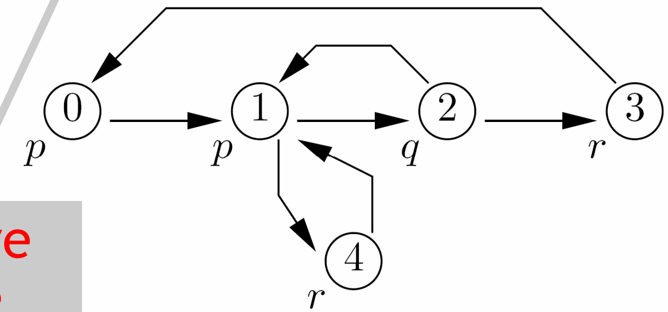
On-the-fly model checking

(illustration on AF^{∞}_{ρ})

Formula: $\mathsf{AF}^{\infty}_{(p|q)^*.r}$

MES: $\left\{ \begin{array}{l} X_0 \stackrel{\nu}{=} X_1 \\ X_1 \stackrel{\nu}{=} \mathsf{AF}_p X_1 \vee \mathsf{AF}_q X_1 \vee \mathsf{AF}_r X_0 \end{array} \right\}$

KS:

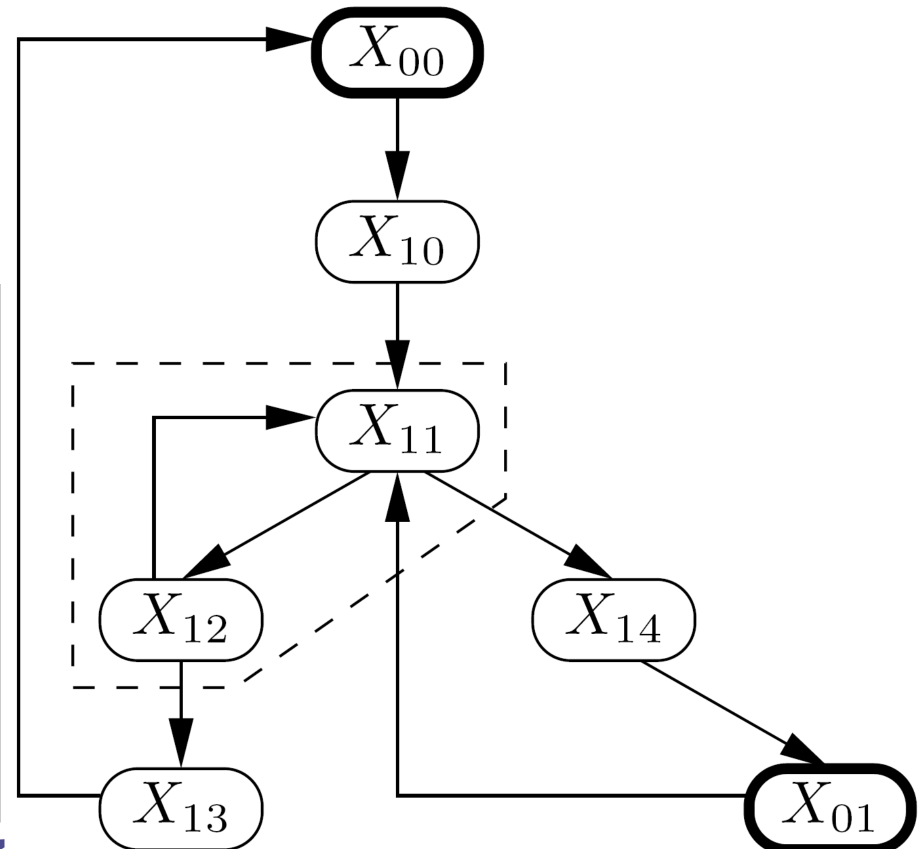


abusive merge of the 2 equation blocks into one ν-block

symmetric version of algorithm $A4_{cyc}$ with marked cycle avoidance

applies for conjunctive ν-blocks obtained from AF^{∞}_{ρ} when ρ is *deterministic*

BES: $X_{ij} = s_j \models X_i$



Complexity of CTRL model checking

Operator	Complexity	
	ρ deterministic	ρ nondeterministic
EF_ρ AG_ρ	$O(\lvert\rho\rvert \cdot (\lvert S\rvert + \lvert T\rvert))$	
AF_ρ EG_ρ	$O(\lvert\rho\rvert \cdot (\lvert S\rvert + \lvert T\rvert))$	$O(2^{\lvert\rho\rvert} \cdot (\lvert S\rvert + \lvert T\rvert))$
EF_ρ^∞ $\mathsf{AG}_\rho^\dashv$	$O(\lvert\rho\rvert \cdot (\lvert S\rvert + \lvert T\rvert))$	
AF_ρ^∞ $\mathsf{EG}_\rho^\dashv$	$O(\lvert\rho\rvert \cdot (\lvert S\rvert + \lvert T\rvert))$	$O(2^{2^{\lvert\rho\rvert}} \cdot (\lvert S\rvert + \lvert T\rvert)^2)$

≈ PDL-delta

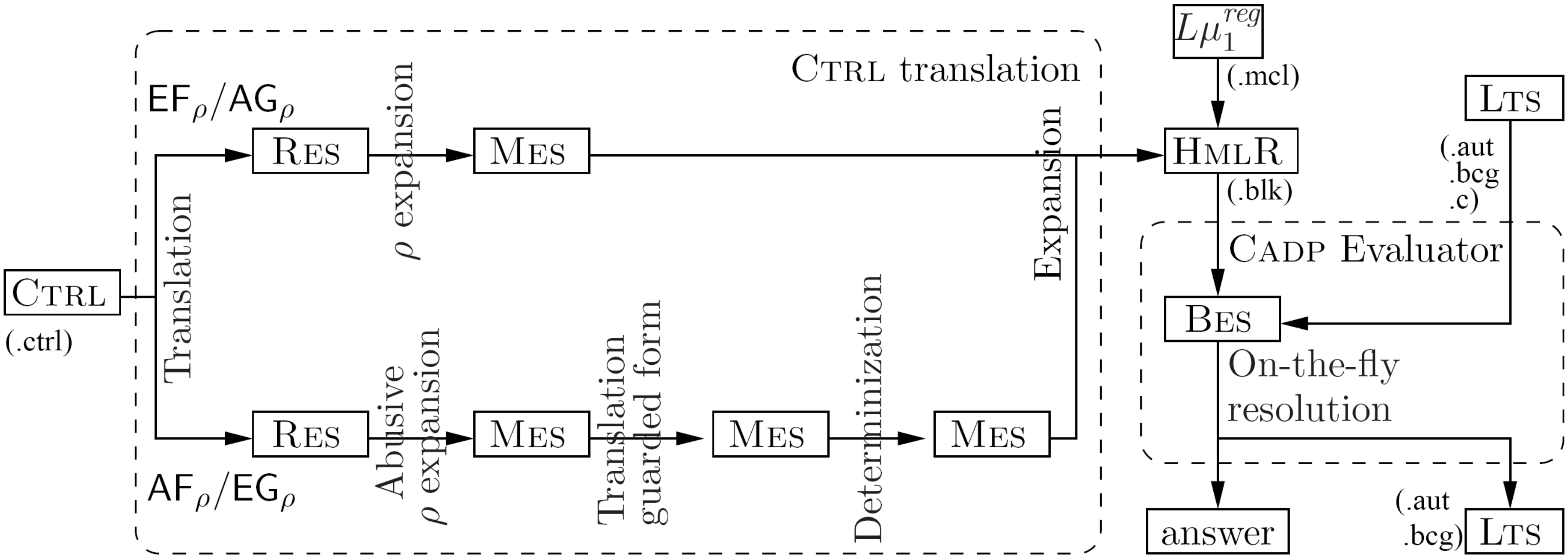
linear-time on-the-fly
model checking
due to A4cyc

(double) exponential in size
of the regular subformula
(but |ρ| << |S|)

quadratic in size
of the KS (general
case of alternation
depth 2)

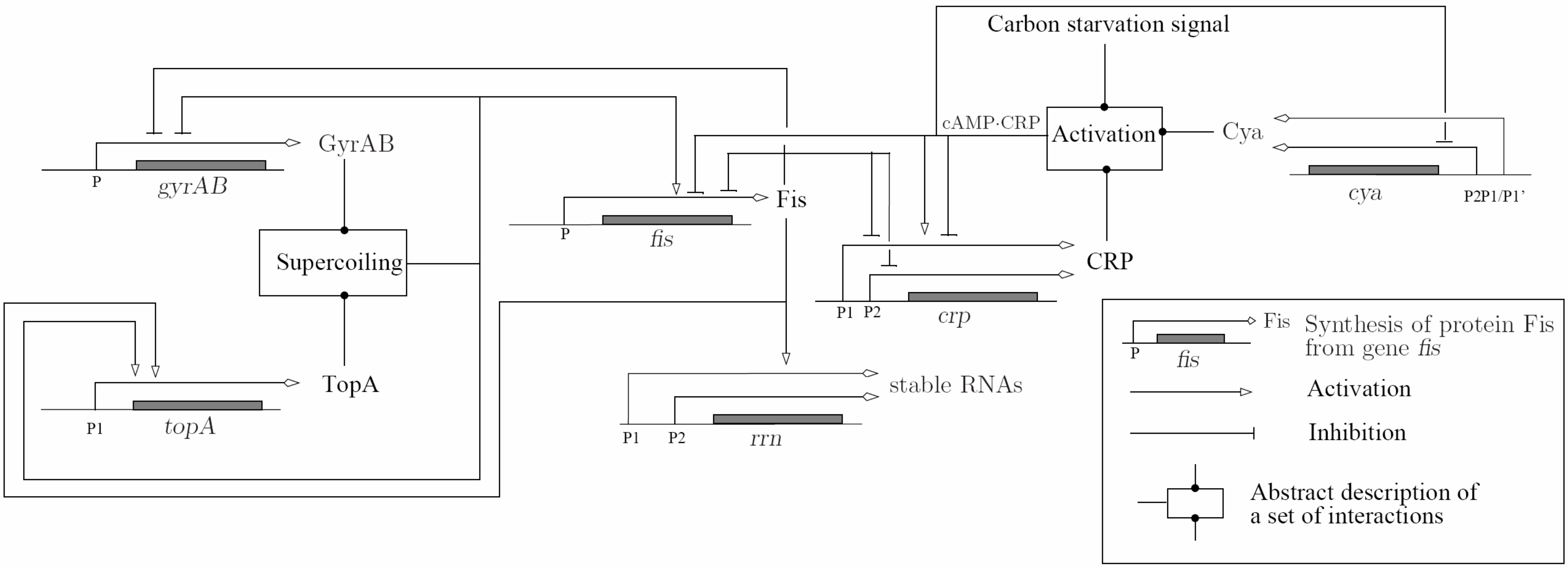
Implementation

- Ctrl2Blk translator (12,000 lines of code)
  - SYNTAX / LOTOS NT compiler construction technology
  - Between 3 and 5 translation phases (including PNF)



Application

- Analysis of the network controlling the carbon starvation response of *E. coli*



Verification of CTRL properties

- Four CTRL properties checked on the state-transition graph of the network:

N.	CTRL formula	Answer
1.	$\mathsf{EF}_{\mathsf{true}^*}\,\mathsf{AF}_{inTermCycle^+.(inc\_rrn^+.dec\_rrn^+)^+}\,\mathsf{true}$	false
2.	$\mathsf{EF}_{\mathsf{true}^*}\,\mathsf{EF}^{\infty}_{inTermCycle^+.(inc\_rrn^+.dec\_rrn^+)^+}$	true
3.	$\mathsf{AG}_{\mathsf{true}^*}\,\mathsf{EF}^{\infty}_{inTermCycle^+.(inc\_rrn^+.dec\_rrn^+)^+}$	false
4.	$\mathsf{AG}_{true^*.inc\_Fis^+.dec\_Crp^+.inTermCycle}\,\mathsf{EF}^{\infty}_{inTermCycle^+.(inc\_rrn^+.dec\_rrn^+)^+}$	true

- Properties unexpressible in CTL or LTL
- Checked in linear-time using Ctrl2Blk + Evaluator

Related work
(extensions of TL with regular constructs)

- LTL
  - ETL [Wolper-82] : LTL + regular grammars
  - ForSpec [Armoni-et-al-02] : LTL + regexps + clocks (HW analysis)
  - Eagle [Barringer-04] : LTL + regexps + rules (runtime verification)
- CTL*
  - CTL* + Büchi automata [Thomas-89]
- CTL
  - BRTL [Hamaguchi-et-al-90]: CTL + deterministic Büchi automata
  - RCTL [Beer-et-al-98] and Sugar [Eisner-et-al-01]: CTL + $AG_\rho\ \varphi$
  - RegCTL [Cerna-01]: CTL + regexps

RCTL $\leq$ RegCTL $\leq$ CTRL

Conclusion and future work

- CTRL (Computation Tree Regular Logic):
  - Combines branching-time and linear-time operators
  - Syntax and semantics definition + translation to HMLR
  - Implementation of Ctrl2Blk translator
  - Connection with the Evaluator 3.6 model checker of CADP

- Ongoing and future work:
  - Local resolution algorithms for BESs with alternation depth two (AF^{∞}_{ρ} for ρ nondeterministic)
  - Static analysis on the GNA atomic propositions
  - Distributed version of the model checker
  - Patterns of biologically-relevant temporal properties