A constraint programming approach to the analysis of Petri nets structural properties and application to biochemical networks

Faten Nabli

Supervised by François Fages and Sylvain Soliman

INRIA Paris-Rocquencourt
CONTRAINTES

July 10, 2013

Outline

Differential equations and structure

A classical way to describe and analyse biochemical reaction systems: Differential equations.

Differential equations and structure

A classical way to describe and analyse biochemical reaction systems: Differential equations.

Reaction model

$$S + E \underset{k_2}{\overset{k_1}{\rightleftharpoons}} ES \xrightarrow{k_3} E + P$$

$$\begin{aligned} dS/dt &= -k_1 \times S \times E + k_2 \times ES \\ dP/dt &= k_3 \times ES \\ dE/dt &= -k_1 \times S \times E + (k_2 + k_3) \times ES \\ dES/dt &= k_1 \times S \times E - (k_2 + k_3) \times ES \end{aligned}$$

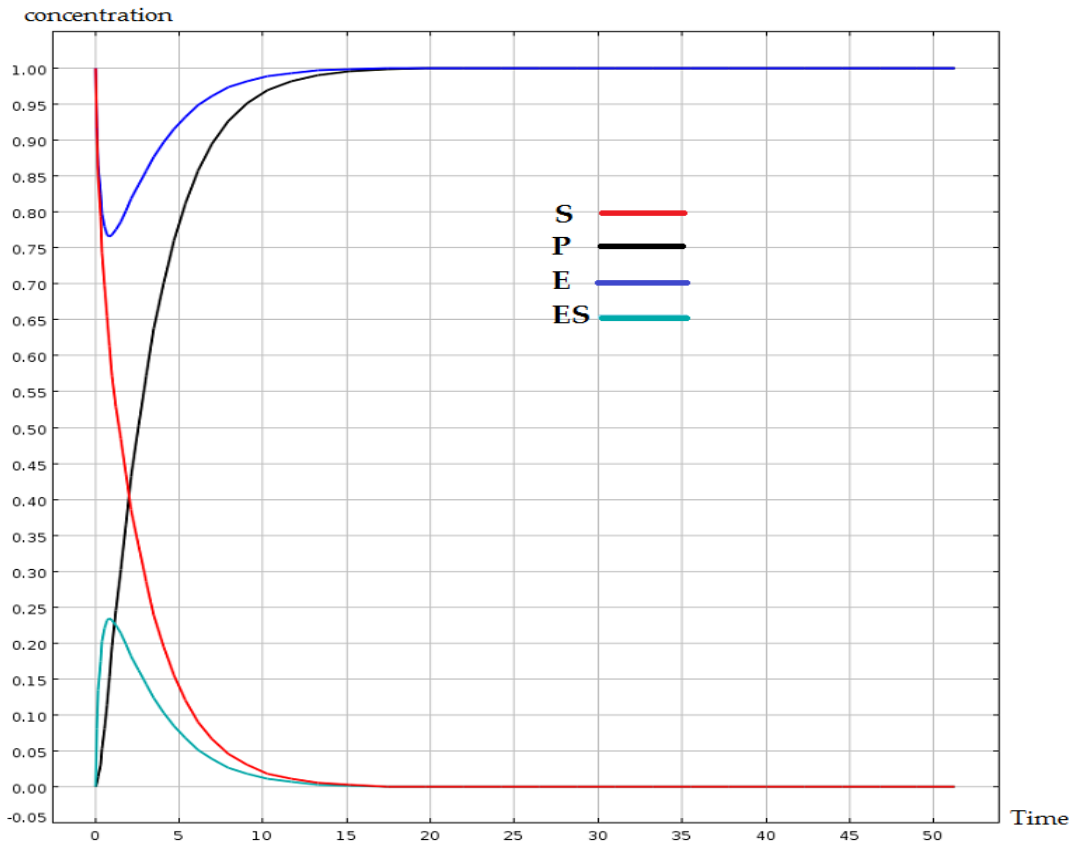
1913 *Die Kinetik der Invertinwirkung*.
L. Menten, M.I. Michaelis. Biochemistry Zeitung 49.

Differential equations and structure

A classical way to describe and analyse biochemical reaction systems: Differential equations.

Reaction model

$$S + E \underset{k_2}{\overset{k_1}{\rightleftharpoons}} ES \xrightarrow{k_3} E + P$$



Differential equations and structure

A classical way to describe and analyse biochemical reaction systems: Differential equations.

Reaction model

$$S + E \underset{k_2}{\overset{k_1}{\rightleftharpoons}} ES \xrightarrow{k_3} E + P$$

$$\begin{aligned} dS/dt &= -k_1 \times S \times E + k_2 \times ES \\ dP/dt &= k_3 \times ES \\ dE/dt &= -k_1 \times S \times E + (k_2 + k_3) \times ES \\ dES/dt &= k_1 \times S \times E - (k_2 + k_3) \times ES \end{aligned}$$

+

Conservation laws

$$E + ES = \text{cte}$$
$$P + S + ES = \text{cte}$$

⇓

Equivalent model:

$$\begin{aligned} dS/dt &= k_2 \times ES - k_1 \times E \times S \\ dES/dt &= k_1 \times E \times S - (k_2 + k_3) \times ES \end{aligned}$$

Differential equations and structure

A classical way to describe and analyse biochemical reaction systems: Differential equations.

Differential equations and structure

A classical way to describe and analyse biochemical reaction systems: Differential equations.

✎ Rate constants usually not known

Differential equations and structure

A classical way to describe and analyse biochemical reaction systems: Differential equations.

- ✎ Rate constants usually not known
- ✎ Analytically intractable even for small systems.

Differential equations and structure

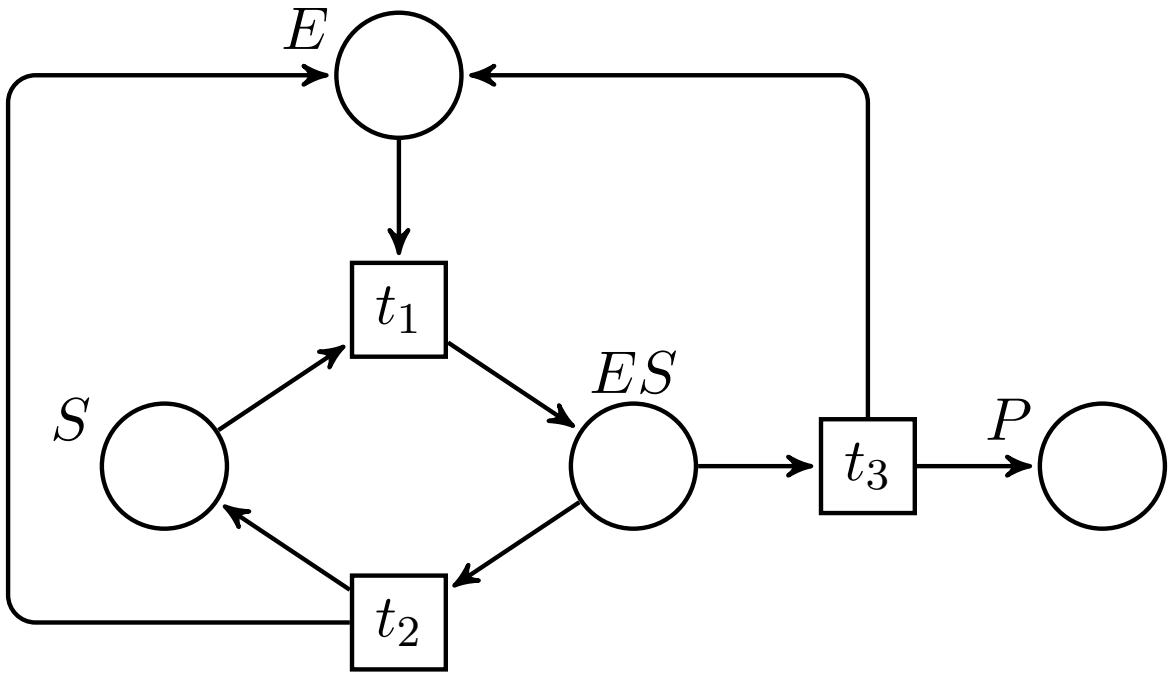
A classical way to describe and analyse biochemical reaction systems: Differential equations.

- ✎ Rate constants usually not known
- ✎ Analytically intractable even for small systems.

Use of Petri nets structural properties to say something about the system dynamics without knowing the kinetics.

Michaelis–Menten enzymatic reactions

Structural model: Reaction graph



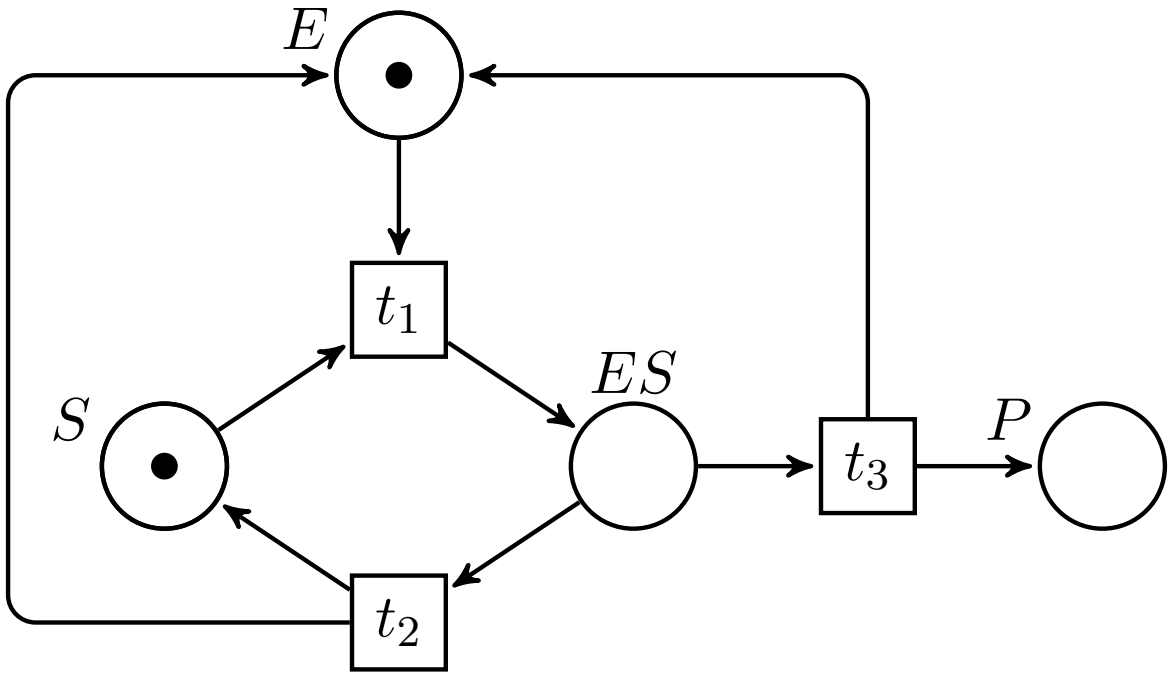
$$S + E \rightleftharpoons ES \longrightarrow E + P$$

Petri-net = reaction graph + discrete dynamics

1962 *Kommunikation mit Automaten.* Carl Adam Petri. Ph. D. Thesis. University of Bonn.

Michaelis–Menten enzymatic reactions

Structural model: Reaction graph



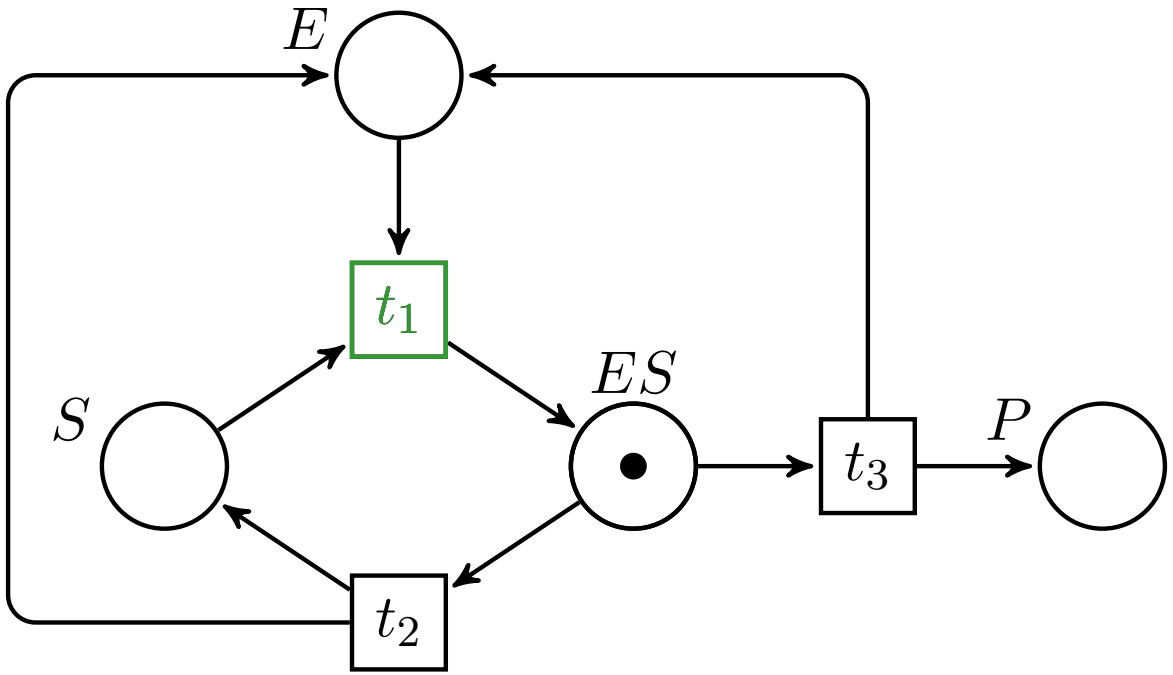
$$S + E \rightleftharpoons ES \longrightarrow E + P$$

Petri-net = reaction graph + discrete dynamics

1962 *Kommunikation mit Automaten.* Carl Adam Petri. Ph. D. Thesis. University of Bonn.

Michaelis–Menten enzymatic reactions

Structural model: Reaction graph



$$S + E \rightleftharpoons ES \longrightarrow E + P$$

Petri-net = reaction graph + discrete dynamics

1962 *Kommunikation mit Automaten.* Carl Adam Petri. Ph. D. Thesis. University of Bonn.

Siphons and traps

Biological interpretation: A siphon (resp. trap) refers to a non-empty set of chemical species that once none (resp. some) of them is present, they will never be produced (resp. disappear at the same time).

2003 *Topological analysis of metabolic networks based on Petri net theory*. I.Zevedei-Oancea and S.Schuster. In Silico Biology.

Siphons and traps

Biological interpretation: A siphon (resp. trap) refers to a non-empty set of chemical species that once none (resp. some) of them is present, they will never be produced (resp. disappear at the same time).

> 2003 *Topological analysis of metabolic networks based on Petri net theory*. I.Zevedei-Oancea and S.Schuster. In Silico Biology.

Dynamical characterisation: A siphon is a non-empty set of places that, once it is unmarked, remains so, whatever the dynamics are.

Structural characterisation: A siphon is a non-empty set of places S such that ${}^\bullet S \subseteq S^\bullet$.

Siphons and traps

Biological interpretation: A siphon (resp. trap) refers to a non-empty set of chemical species that once none (resp. some) of them is present, they will never be produced (resp. disappear at the same time).

> 2003 *Topological analysis of metabolic networks based on Petri net theory*. I.Zevedei-Oancea and S.Schuster. In Silico Biology.

Dynamical characterisation: A siphon is a non-empty set of places that, once it is unmarked, remains so, whatever the dynamics are.

Structural characterisation: A siphon is a non-empty set of places S such that ${}^\bullet S \subseteq S^\bullet$.
A siphon is minimal if it does not contain any other siphon.

Siphons/Traps and Computation Tree Logic

CTL is used for verifying temporal properties by model checking.

2003 *Symbolic model checking of biochemical networks*.
Nathalie Chabrier and François Fages. CMSB.

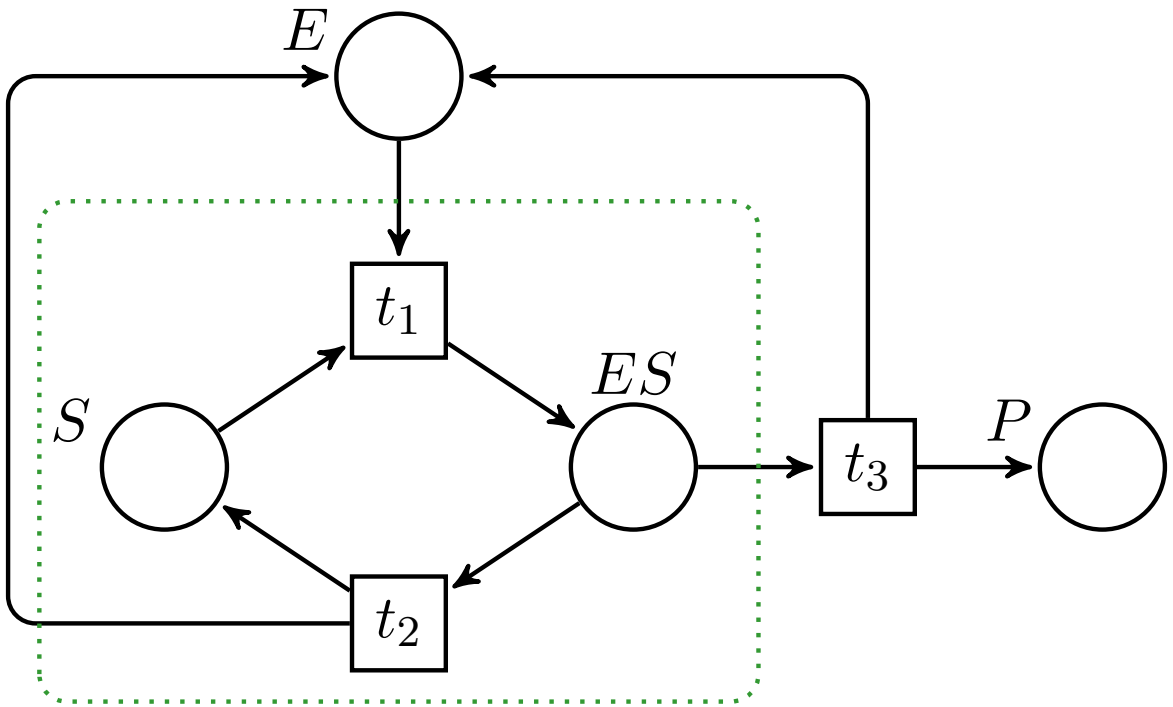
http://contraintes.inria.fr/biocham

Definition. Given a marking m, for any subset $P' \subseteq P$ of places, let $\phi_{P'} = \bigvee_{p \in P'} m_p \geq 1$.

Theorem. The set $P' \subseteq P$ is a trap if and only if for all $s \in \mathbb{N}^P$, $(\mathcal{S}^{PN}, s) \models \phi_{P'} \Rightarrow AG\phi_{P'}$.

Theorem. The set $P' \subseteq P$ is a siphon if and only if for all $s \in \mathbb{N}^P$, $(\mathcal{S}^{PN}, s) \models \neg\phi_{P'} \Rightarrow AG\neg\phi_{P'}$.

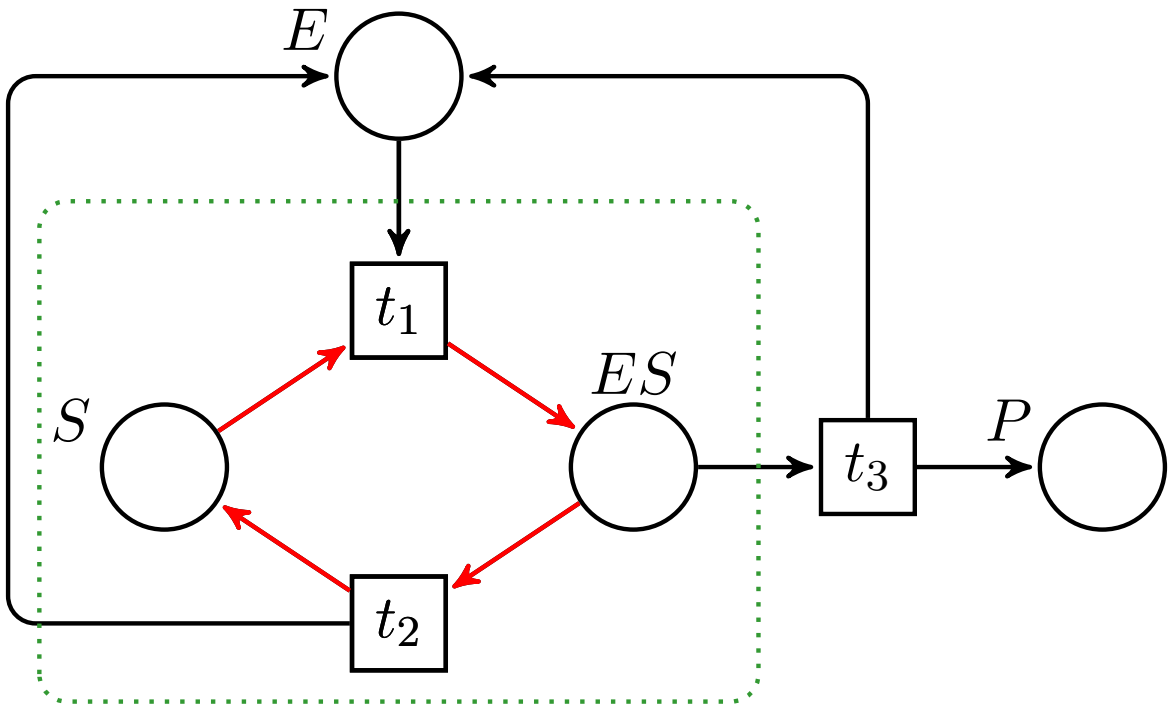
Minimal siphon example



$$^\bullet\{S, ES\} = \{t_1, t_2\} \quad \{S, ES\}^\bullet = \{t_1, t_2, t_3\}$$

S siphon iff $^\bullet S \subseteq S^\bullet$

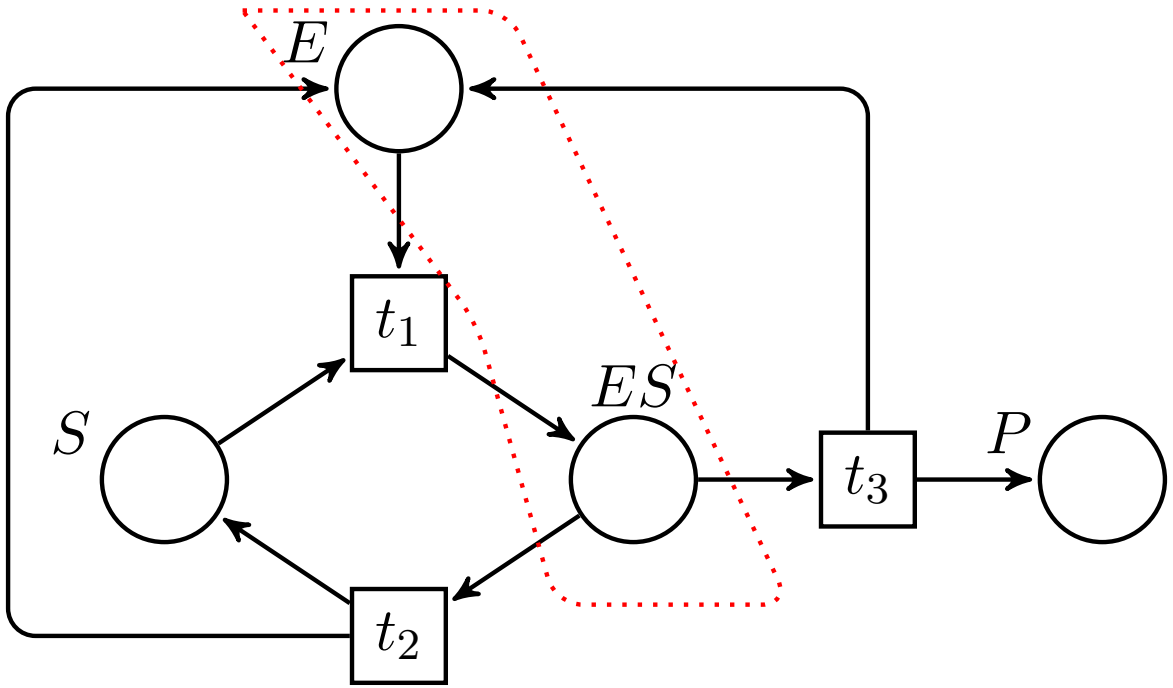
Minimal siphon example



$$^\bullet\{S, ES\} = \{t_1, t_2\} \quad \{S, ES\}^\bullet = \{t_1, t_2, t_3\}$$

S siphon iff $^\bullet S \subseteq S^\bullet$

Minimal trap example



$$\{E, ES\}^\bullet = \{t_1, t_2, t_3\} \quad {}^\bullet\{E, ES\} = \{t_1, t_2, t_3\}$$

D trap iff $D^\bullet \subseteq {}^\bullet D$

Siphon-Trap Property (STP)

Definition. The STP holds when every siphon includes a marked trap.

Theorem. An ordinary Petri net in which the STP holds is deadlock-free.

2010 *On the importance of the deadlock trap property for monotonic liveness.*
M. Heiner, C. Mahulea, and M. Silva. In Int. Workshop on Biological Processes and Petri Nets (BioPPN).

Siphon-Trap Property (STP)

Definition. The STP holds when every siphon includes a marked trap.

Theorem. An ordinary Petri net in which the STP holds is deadlock-free.

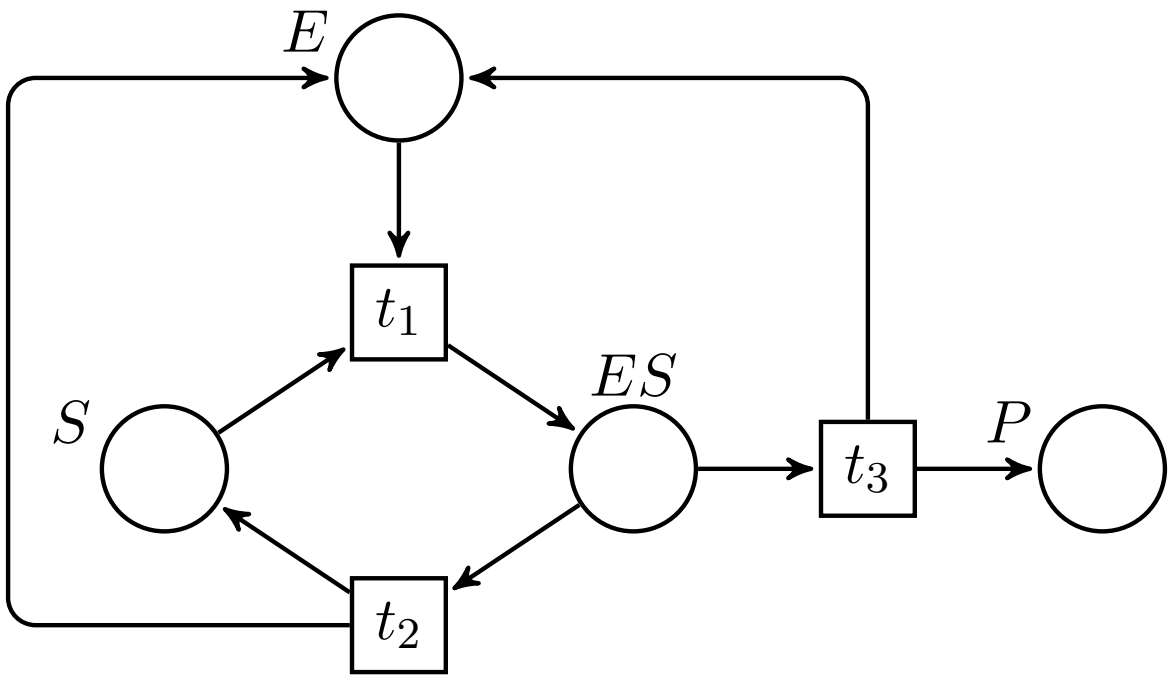
2010 *On the importance of the deadlock trap property for monotonic liveness.*
M. Heiner, C. Mahulea, and M. Silva. In Int. Workshop on Biological Processes and Petri Nets (BioPPN).

2010 *New Algorithms for Deciding the Siphon-Trap Property.*
O. Oanea, H. Wimmel, and K. Wolf. Petri nets 2010.

P-invariant: Biological interpretation

A P-invariant corresponds to a conservation law.

1993 *Petri net representations in metabolic pathways.* V. N. Reddy, M. L. Mavrovouniotis and M. N. Liebman. ISMB.

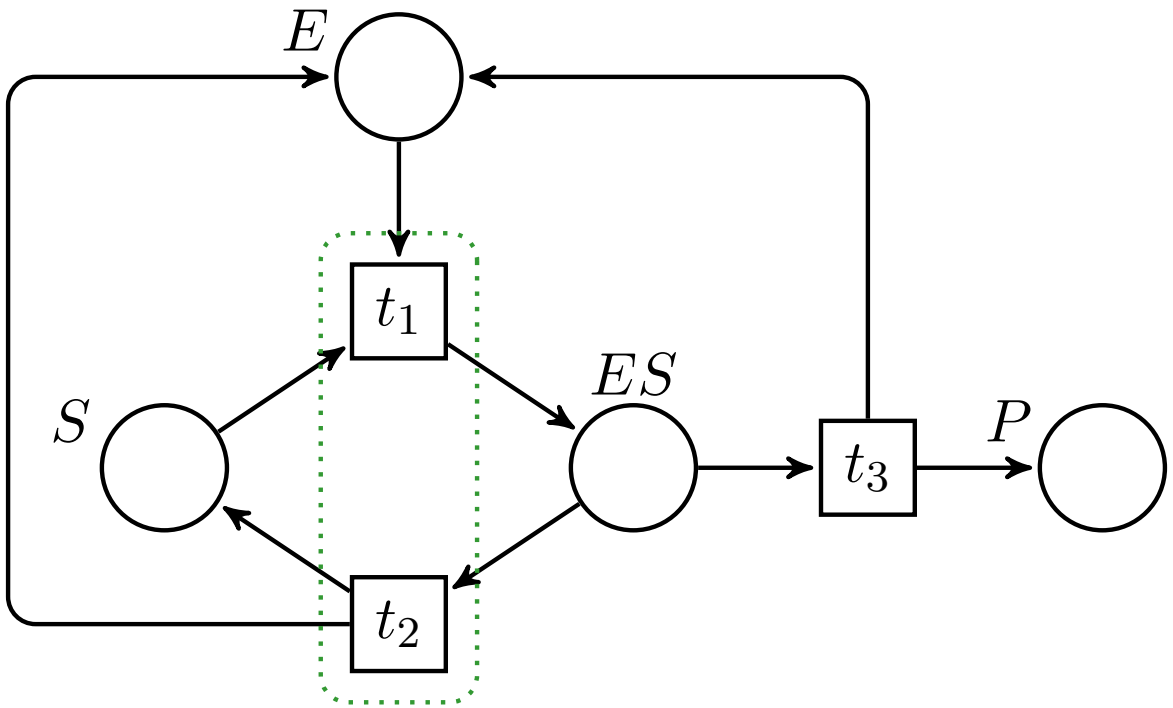


$$E + ES = \text{cte}$$
$$P + S + ES = \text{cte}$$

T-invariants: Biological interpretation

Minimal T-invariants correspond to elementary flux modes or reversible reactions.

1994 *On elementary flux modes in biochemical reaction systems at steady state*. S. Schuster and C. Hilgetag. Biol. Syst.

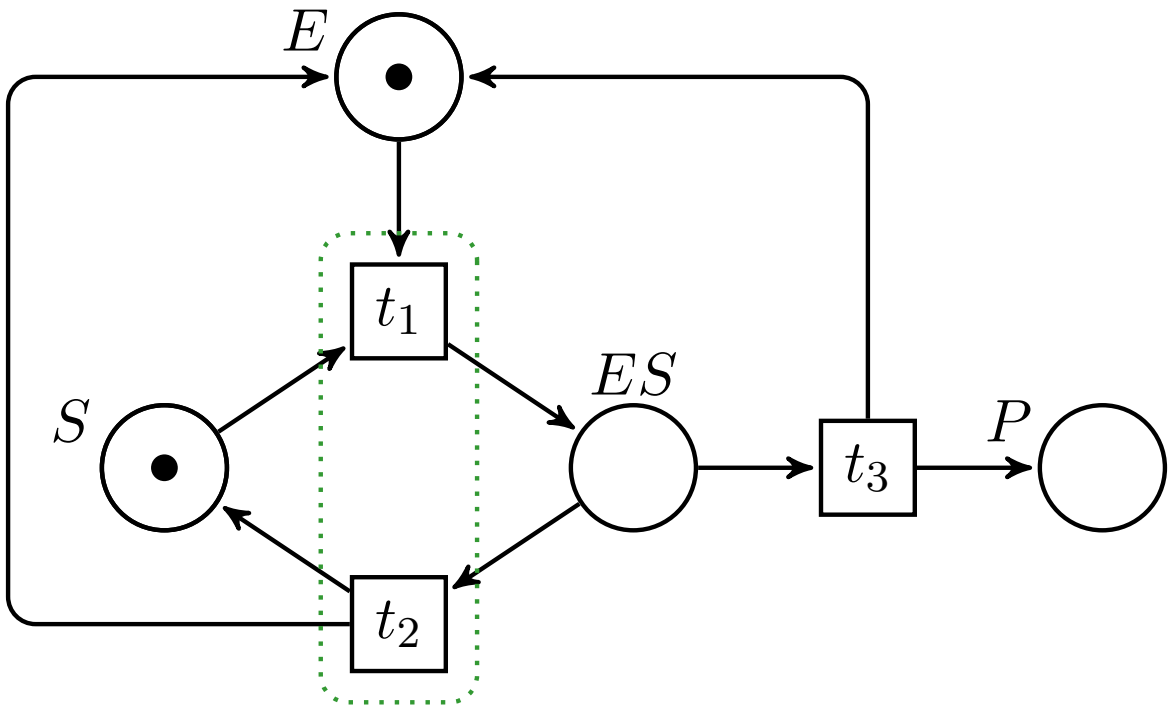


$$t_1 = t_2 = 1 \wedge t_3 = 0$$

T-invariants: Biological interpretation

Minimal T-invariants correspond to elementary flux modes or reversible reactions.

1994 *On elementary flux modes in biochemical reaction systems at steady state*. S. Schuster and C. Hilgetag. Biol. Syst.

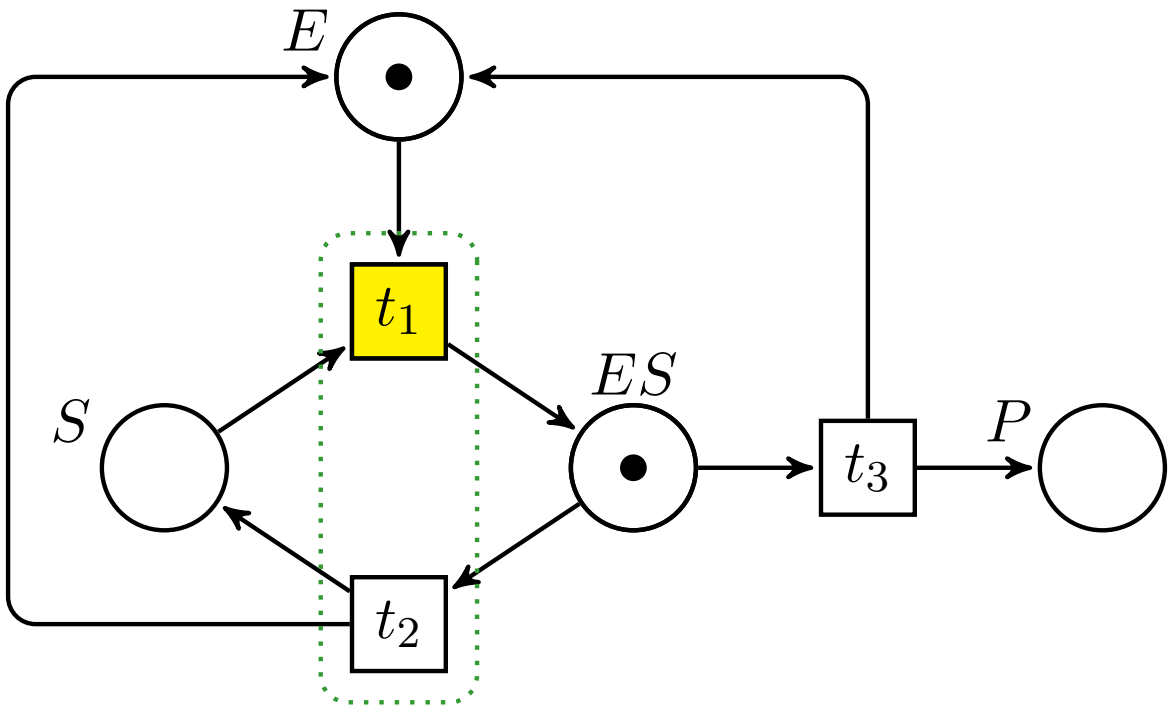


$$t_1 = t_2 = 1 \wedge t_3 = 0$$

T-invariants: Biological interpretation

Minimal T-invariants correspond to elementary flux modes or reversible reactions.

1994 *On elementary flux modes in biochemical reaction systems at steady state*. S. Schuster and C. Hilgetag. Biol. Syst.

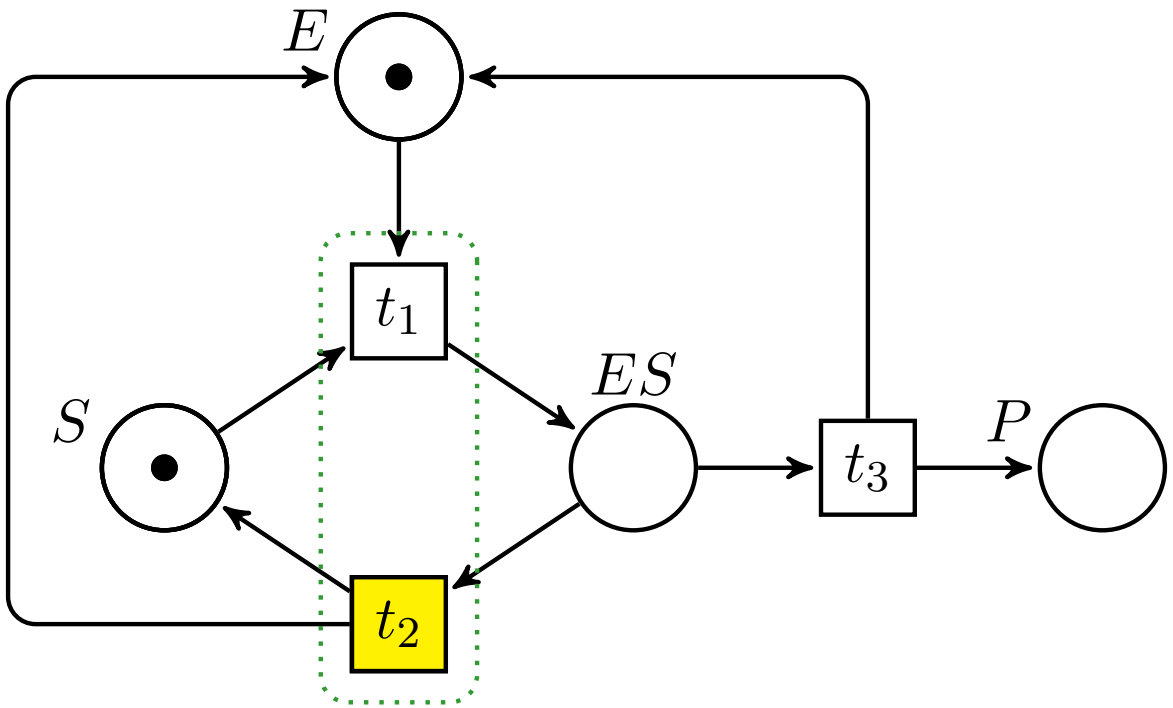


$$t_1 = t_2 = 1 \wedge t_3 = 0$$

T-invariants: Biological interpretation

Minimal T-invariants correspond to elementary flux modes or reversible reactions.

1994 *On elementary flux modes in biochemical reaction systems at steady state*. S. Schuster and C. Hilgetag. Biol. Syst.

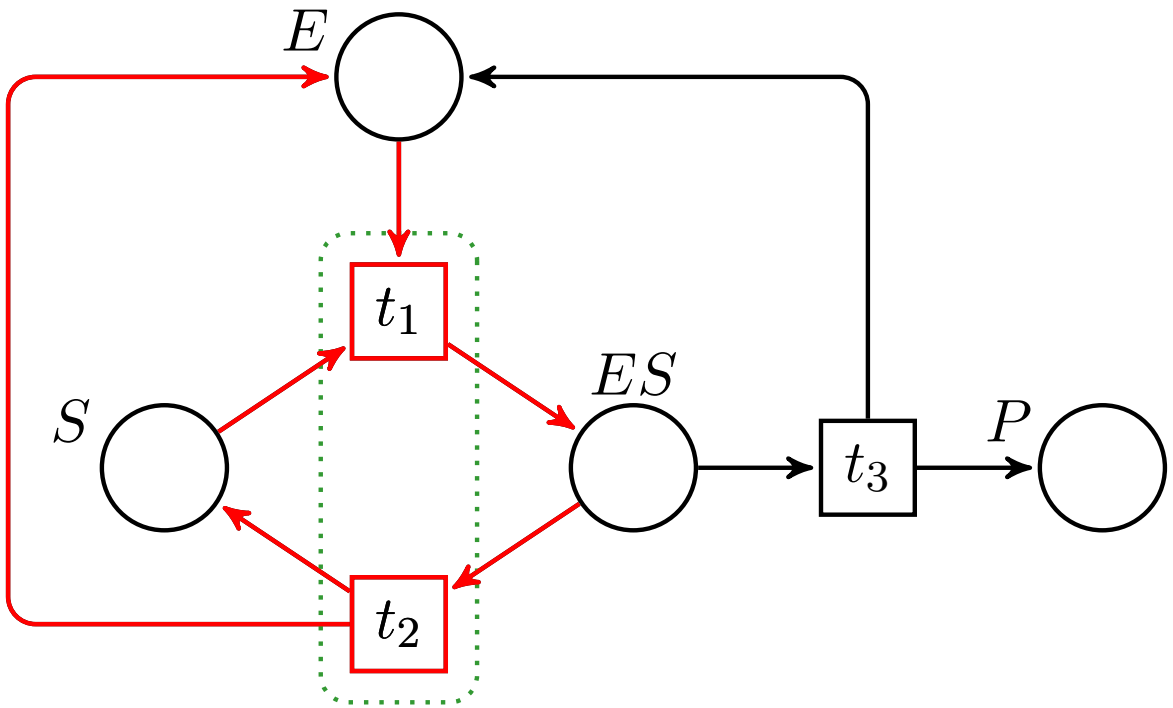


$$t_1 = t_2 = 1 \wedge t_3 = 0$$

T-invariants: Biological interpretation

Minimal T-invariants correspond to elementary flux modes or reversible reactions.

1994 *On elementary flux modes in biochemical reaction systems at steady state*. S. Schuster and C. Hilgetag. Biol. Syst.



$$t_1 = t_2 = 1 \wedge t_3 = 0$$

Benchmark for evaluation

Database **Biomodels.net** (version March 2012)

404 manually curated quantitative biochemical models.
Average ~ 50 species, ~ 90 reactions.
Biggest model has 194 species, 313 reactions.
Reference publication for each model.

> 2006 *BioModels Database: a free, centralized database of curated, published, quantitative kinetic models of biochemical and cellular systems.*
> le Novère et al. Nucleic Acid Research.

Benchmark for evaluation (2)

Database **Petriweb**

Repository of 80 models modelling real industrial processes.
Average ~ 10 places, ~ 8 transitions.
Biggest model has 68 places, 51 transitions.

2006 *Petriweb: A Repository for Petri Nets*, R. Goud et al. Petri Nets and Other Models of Concurrency - ICATPN.

Complexity

Our challenge: enumerate all minimal siphons in Biomodels.net and Petriweb databases.

Complexity

Our challenge: enumerate all minimal siphons in Biomodels.net and Petriweb databases.

- There can be an exponential number of minimal siphons and minimal T-invariants.

Complexity

Our challenge: enumerate all minimal siphons in Biomodels.net and Petriweb databases.

- There can be an exponential number of minimal siphons and minimal T-invariants.
- Deciding the existence of a minimal siphon of a given cardinality in a general Petri net is NP-complete.

1996 *Finding minimal siphons in general Petri nets.*
S. Tanimoto, M. Yamauchi, and T. Watanabe. IEICE.

Complexity

Our challenge: enumerate all minimal siphons in Biomodels.net and Petriweb databases.

- There can be an exponential number of minimal siphons and minimal T-invariants.
- Deciding the existence of a minimal siphon of a given cardinality in a general Petri net is NP-complete.

  1996 *Finding minimal siphons in general Petri nets.* S. Tanimoto, M. Yamauchi, and T. Watanabe. IEICE.

- Deciding the existence of a minimal siphon containing a given set of places in a general Petri net is NP-complete.

  1999 *Time complexity analysis of the minimal siphon extraction problem of petri nets.* S. Tanimoto, M. Yamauchi, and T. Watanabe. IEICE.

Thesis contribution

- Finding minimal siphons and minimal T-invariants as a constraint satisfaction problem.

Thesis contribution

- Finding minimal siphons and minimal T-invariants as a constraint satisfaction problem.
- Enumerating all minimal siphons with a Customized branch & bound procedure.

Thesis contribution

- Finding minimal siphons and minimal T-invariants as a constraint satisfaction problem.
- Enumerating all minimal siphons with a Customized branch & bound procedure.
- Successful enumeration of all minimal siphons on all Biomodels.net and Petriweb.

Thesis contribution

- Finding minimal siphons and minimal T-invariants as a constraint satisfaction problem.
- Enumerating all minimal siphons with a Customized branch & bound procedure.
- Successful enumeration of all minimal siphons on all Biomodels.net and Petriweb.
- Outperforming state-of-the-art algorithms on Biomodels.net and Petriweb.

Thesis contribution

- Finding minimal siphons and minimal T-invariants as a constraint satisfaction problem.
- Enumerating all minimal siphons with a Customized branch & bound procedure.
- Successful enumeration of all minimal siphons on all Biomodels.net and Petriweb.
- Outperforming state-of-the-art algorithms on Biomodels.net and Petriweb.
- Towards understanding why our techniques are efficient: Linear time complexity on classes of Petri nets of bounded tree-widths.

Boolean model for siphons and traps

Boolean Model of Siphons

variables

$$(\forall p)\ X_p = 1 \Longleftrightarrow p \in S$$

constraints

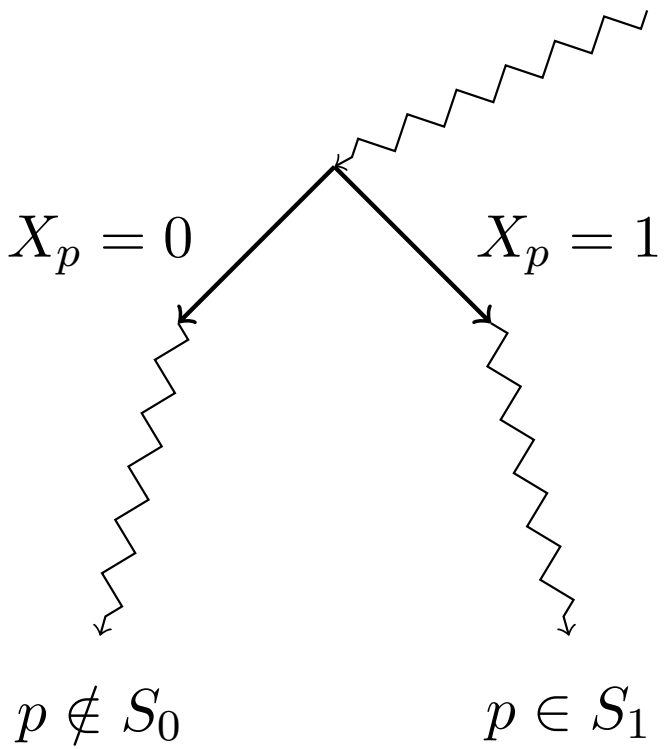
$$(\forall p)\ X_p = 1 \Rightarrow \bigwedge_{t \in {}^\bullet p} \bigvee_{p' \in {}^\bullet t} X_{p'} = 1$$

Finding siphons is reduced to finding

Boolean assignments satisfying these formulas.

Search strategy ensuring minimality

Try absence 0 then presence 1



0 before 1 and S_0 before S_1 in the search tree $\Longrightarrow S_1 \not\subseteq S_0$

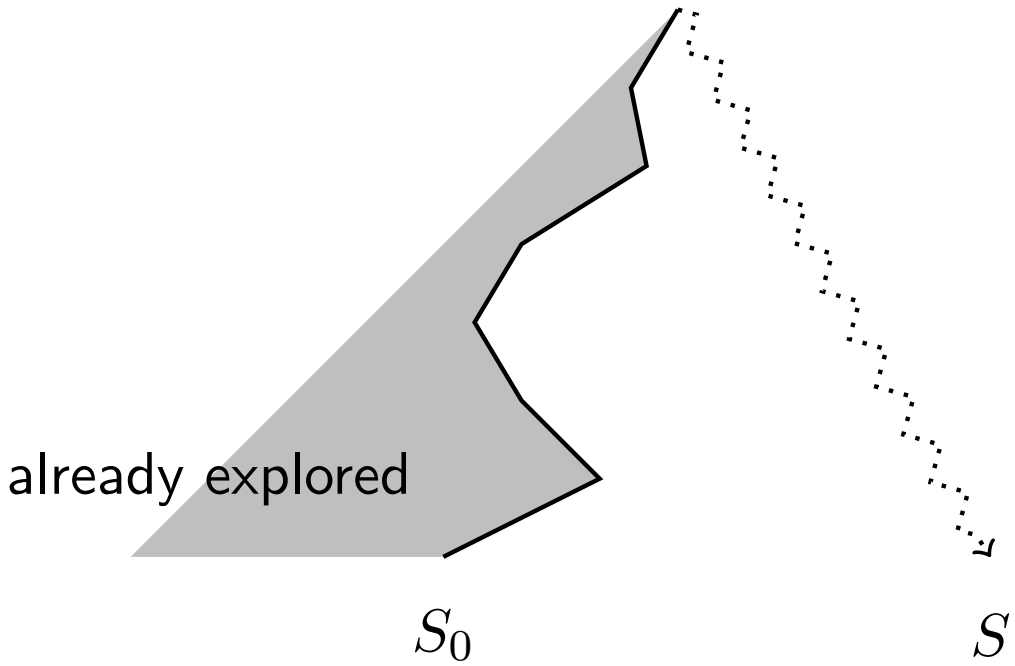
Enumerating all minimal siphons by iteration

Once a (minimal) siphon S is found

1. add the constraint $\bigvee_{p \in S} X_p = 0$
2. restart the search

Optimisation of the search in CLP($\mathcal{B}$)

"Those who cannot remember the past are condemned to repeat it"
George Santayana



2 possible strategies:

add constraint $S \geqslant_{\text{lex}} S_0$
replay search procedure: most efficient!

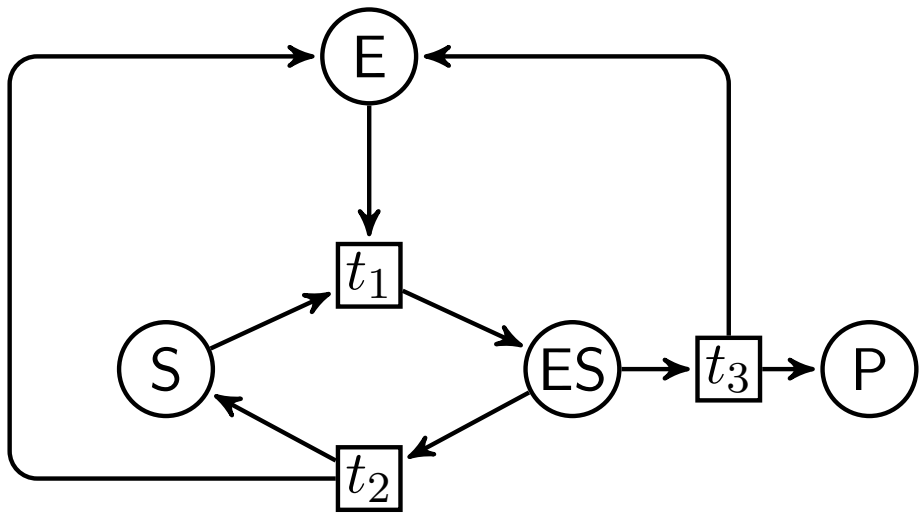
Example: Michaelis-Menten enzymatic reaction

Boolean variables

e, s, es and p

initial clauses

$$e \vee s \vee es \vee p$$
$$\neg es \vee e \vee s$$
$$\neg s \vee es$$
$$\neg e \vee es$$
$$\neg e \vee es$$
$$\neg p \vee es$$



Example: Michaelis-Menten enzymatic reaction

Boolean variables

$$e,\ s,\ es \text{ and } p$$

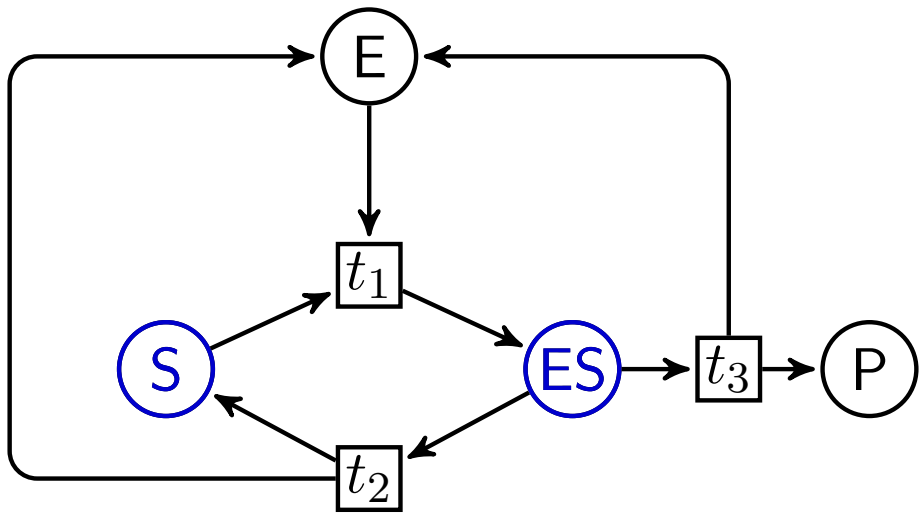
assignment

$$s = es = 1 \wedge e = p = 0$$

$\Rightarrow \{S, ES\}$ is a minimal siphon.

minimality constraint:

$$\neg s \vee \neg es$$



Example: Michaelis-Menten enzymatic reaction

Boolean variables

$$e,\ s,\ es \text{ and } p$$

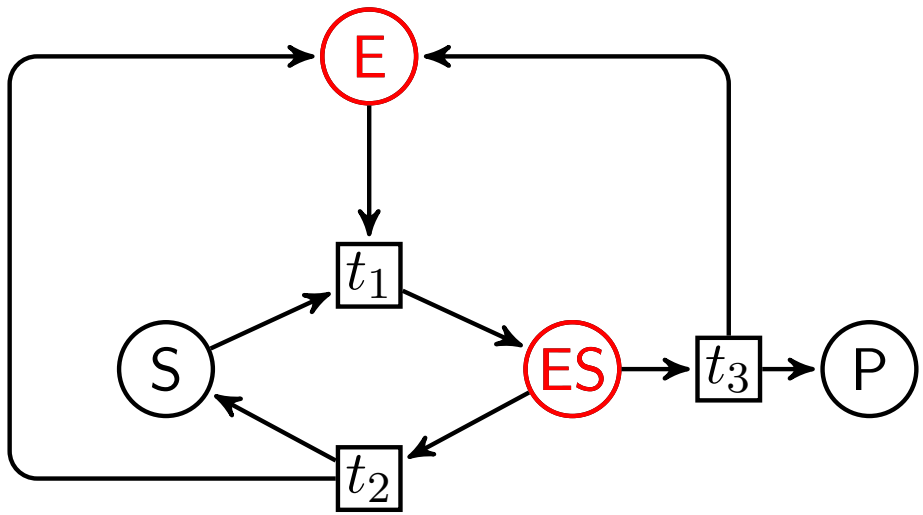
second iteration. assignment:

$$e = es = 1 \wedge s = p = 0$$

$\Rightarrow$ $\{E, ES\}$ is also a minimal siphon.

minimality constraint:

$$\neg e \vee \neg es$$

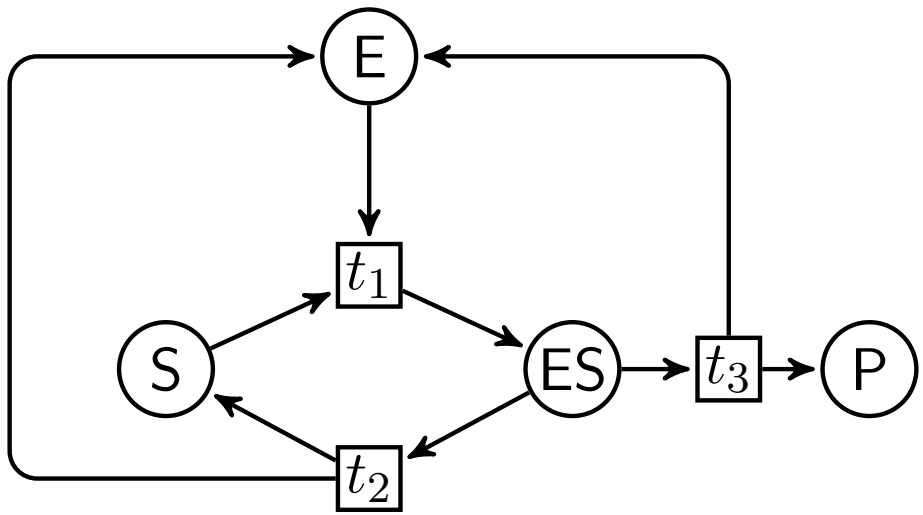


Example: Michaelis-Menten enzymatic reaction

Boolean variables

e, s, es and p

No more variable assignment.
2 minimal siphons: $\{S, ES\}$ and $\{E, ES\}$.



State-of-the-art algorithms for enumerating minimal siphons

2002 *Characterization of minimal and basis siphons with predicate logic and binary programming.* R. Cordone, L. Ferrarini, and L. Piroddi. IEEE CACSD.

2005 *Enumeration algorithms for minimal siphons in Petri nets based on place constraints.* R. Cordone, L. Ferrarini, and L. Piroddi. IEEE TSC.

State-of-the-art algorithms for enumerating minimal siphons

2002 *Characterization of minimal and basis siphons with predicate logic and binary programming.* R. Cordone, L. Ferrarini, and L. Piroddi. IEEE CACSD.

2005 *Enumeration algorithms for minimal siphons in Petri nets based on place constraints.* R. Cordone, L. Ferrarini, and L. Piroddi. IEEE TSC.

PN size	#minimal siphons (avg)	total time (in s.) MIP 2002	dedicated 2005
5	2	0.03	0.05
10	10	0.28	0.07
15	60	5.45	0.39
20	302	303.47	6.84

State-of-the-art algorithms for enumerating minimal siphons

2002 *Characterization of minimal and basis siphons with predicate logic and binary programming.* R. Cordone, L. Ferrarini, and L. Piroddi. IEEE CACSD.

2005 *Enumeration algorithms for minimal siphons in Petri nets based on place constraints.* R. Cordone, L. Ferrarini, and L. Piroddi. IEEE TSC.

PN size	#minimal siphons (avg)	total time (in s.) MIP 2002	dedicated 2005
5	2	0.03	0.05
10	10	0.28	0.07
15	60	5.45	0.39
20	302	303.47	6.84

2012 *Computation of all minimal siphons in Petri nets* S.G. Wang, Y. Li, C.Y. Wang, M.C. Zhou. ICNSC.

2013 *Computation of Minimal Siphons in Petri Nets by Using Binary Decision Diagrams* Y. Chen, G. Liu. ACM-TECS

Enumerating minimal siphons with SAT and CLP($\mathcal{B}$)

database	#models	#P,#T (avg)	#P+#T (max)	#siphons (avg)	total time (in ms.) algo. 2005	mini SAT	GNU-Prolog
Petriweb	80	10,8	119	2.85	2325	156	6

Enumerating minimal siphons with SAT and CLP($\mathcal{B}$)

database	#models	#P,#T (avg)	#P+#T (max)	#siphons (avg)	total time (in ms.) algo. 2005	mini SAT	GNU-Prolog
Petriweb	80	10,8	119	2.85	2325	156	6
Biomodels.net	403	50,90	507	4.21	19734	611	195

Enumerating minimal siphons with SAT and CLP($\mathcal{B}$)

database	#models	#P,#T (avg)	#P+#T (max)	#siphons (avg)	total time (in ms.) algo. 2005	mini SAT	GNU-Prolog
Petriweb	80	10,8	119	2.85	2325	156	6
Biomodels.net	403	50,90	507	4.21	19734	611	195
Biomodel #175	-	118, 194	312	3042	∞	137000	∞

Enumerating minimal siphons with SAT and CLP($\mathcal{B}$)

database	#models	#P,#T (avg)	#P+#T (max)	#siphons (avg)	total time (in ms.) algo. 2005	mini SAT	GNU-Prolog
Petriweb	80	10,8	119	2.85	2325	156	6
Biomodels.net	403	50,90	507	4.21	19734	611	195
Biomodel #175	-	118, 194	312	3042	∞	137000	∞
Kohn's map	-	509, 775	1284	81	28	1	221

Enumerating minimal siphons with SAT and CLP($\mathcal{B}$)

database	#models	#P,#T (avg)	#P+#T (max)	#siphons (avg)	total time (in ms.) algo. 2005	mini SAT	GNU-Prolog
Petriweb	80	10,8	119	2.85	2325	156	6
Biomodels.net	403	50,90	507	4.21	19734	611	195
Biomodel #175	-	118, 194	312	3042	∞	137000	∞
Kohn's map	-	509, 775	1284	81	28	1	221

Enumerating minimal siphons with SAT and CLP($\mathcal{B}$)

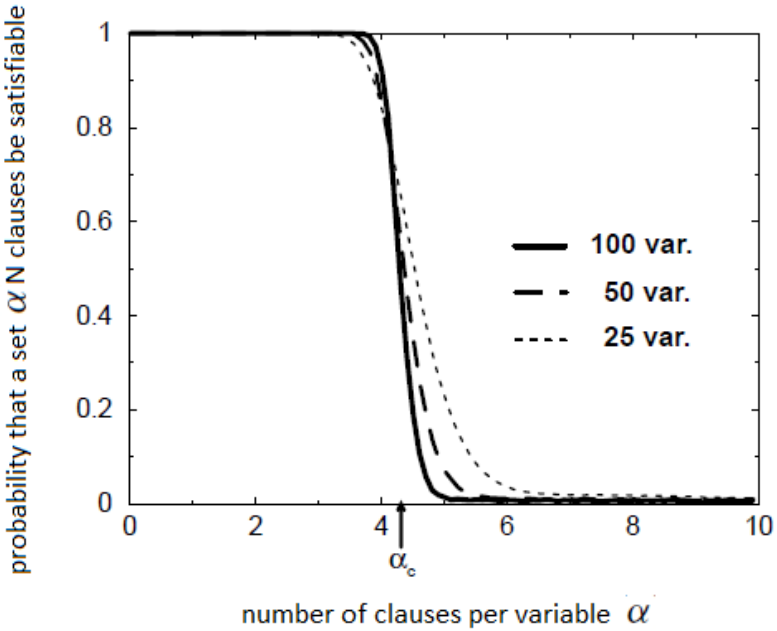
database	#models	#P,#T (avg)	#P+#T (max)	#siphons (avg)	total time (in ms.) algo. 2005	mini SAT	GNU-Prolog
Petriweb	80	10,8	119	2.85	2325	156	6
Biomodels.net	403	50,90	507	4.21	19734	611	195
Biomodel #175	-	118, 194	312	3042	∞	137000	∞
Kohn's map	-	509, 775	1284	81	28	1	221

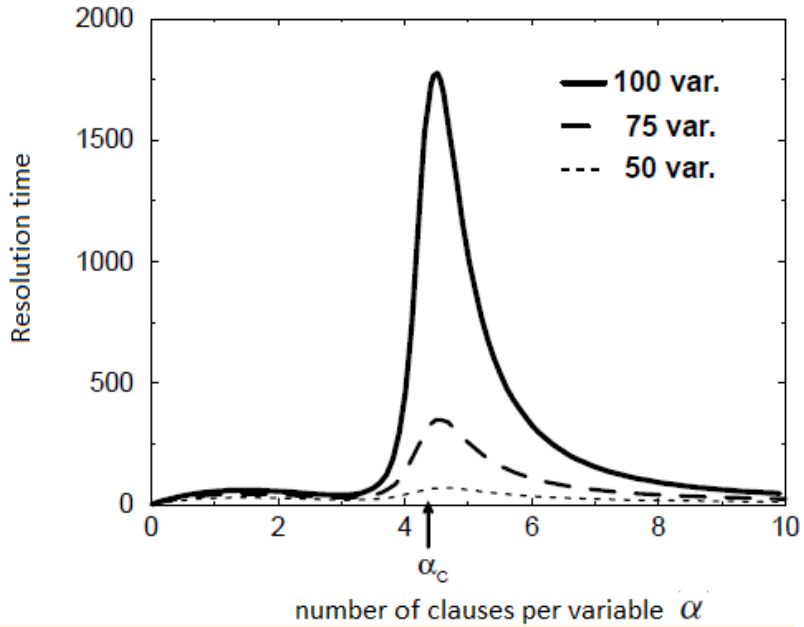
but why are we so efficient?

3-SAT phase transition

$$\alpha = \frac{\#\text{clauses}}{\#\text{variables}}$$

Phase transition for satisfiability at $\alpha \simeq 4.26$



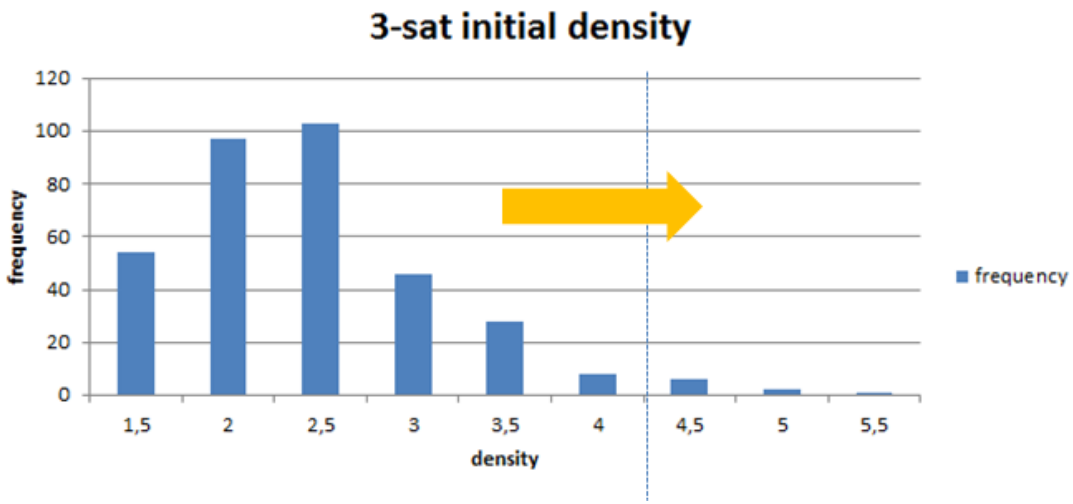


1996 *Generating Hard Satisfiability Problems.* B. Selman, D. Mitchell and H. Levesque. Artificial Intelligence.

Initial density $\geq$ 4.2 ?

$$\text{initial density}_{\text{3-SAT}} = \frac{\sum_{t \in T} \mid t^{\bullet} \mid +1+\mu}{\mid \text{places} \mid +\mu} \text{ where}$$

$$\mu = \sum_{t \in T} \max(0, \mid {}^{\bullet}t \mid -2)$$



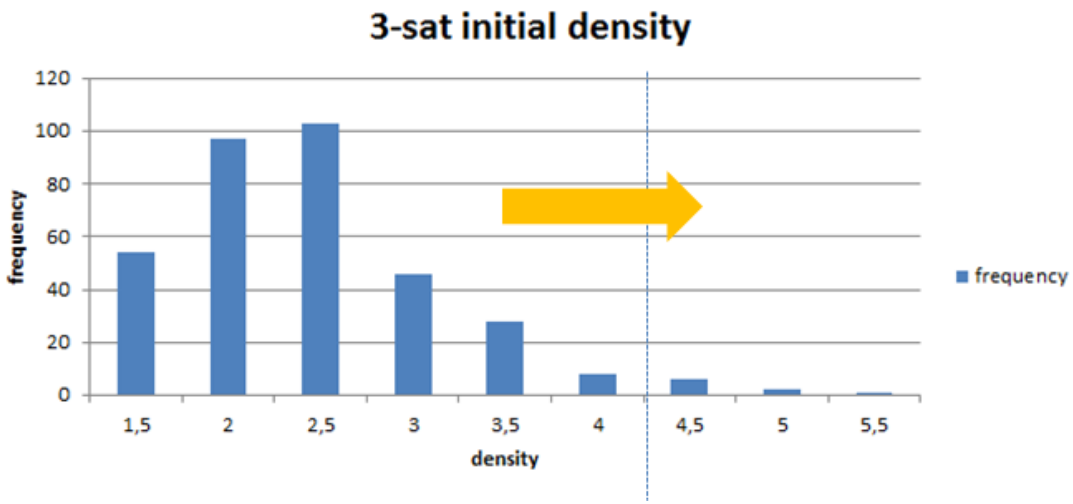
3-SAT initial density of BIOMD175=2.39

The density grows during enumeration: potentially hard problems.

Initial density ≥ 4.2 ?

$$\text{initial density}_{\text{3-SAT}} = \frac{\sum_{t \in T} \mid t^{\bullet} \mid +1+\mu}{\mid \text{places} \mid +\mu} \text{ where}$$

$$\mu = \sum_{t \in T} \max(0, \mid {}^{\bullet}t \mid -2)$$



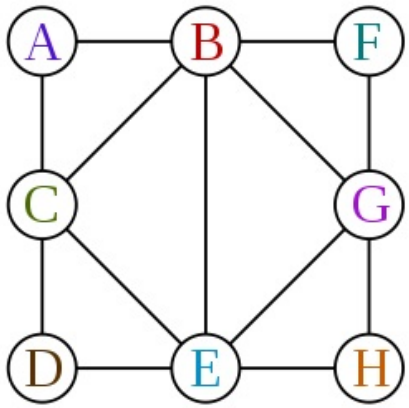
3-SAT initial density of BIOMD175=2.39

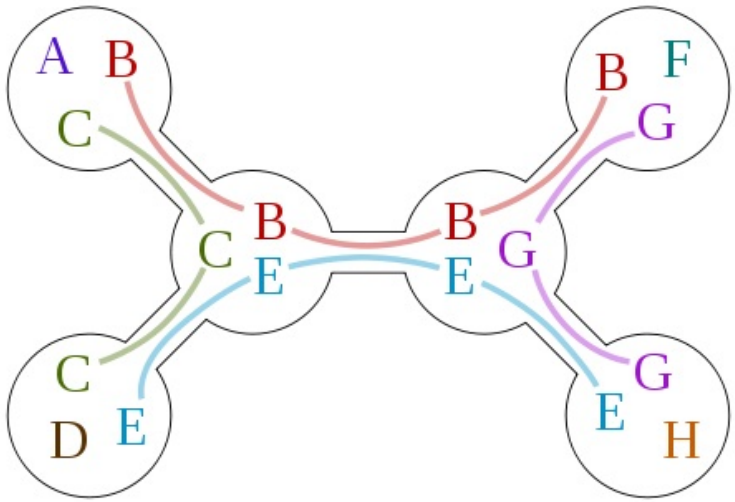
The density grows during enumeration: potentially hard problems.
Not sufficient to explain good performances.

Tree-width of non-oriented graph

Tree-decomposition of a graph:

1. Each graph vertex is associated with at least one tree node.

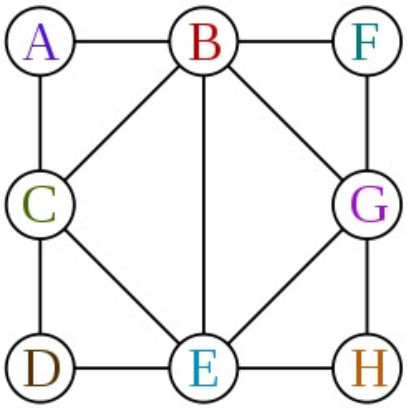


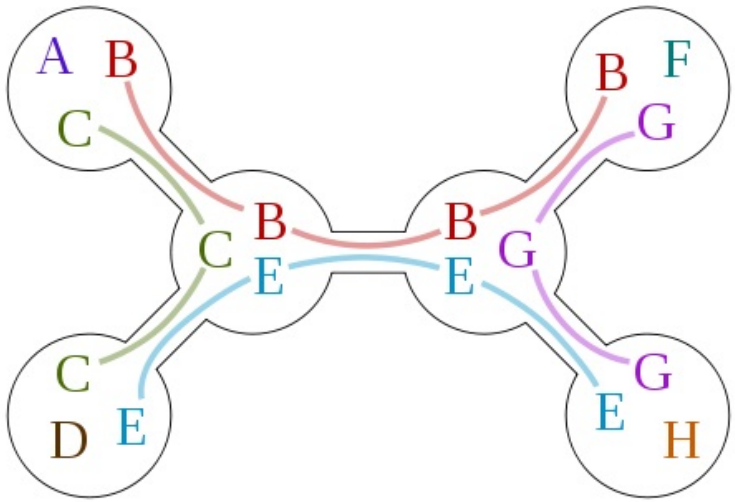


Tree-width of non-oriented graph

Tree-decomposition of a graph:

1. Each graph vertex is associated with at least one tree node.
2. Each graph edge connects two vertices that are listed together at some tree node.

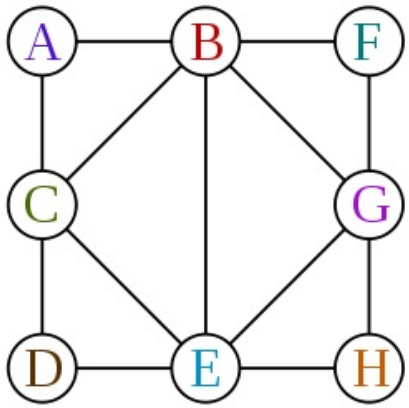


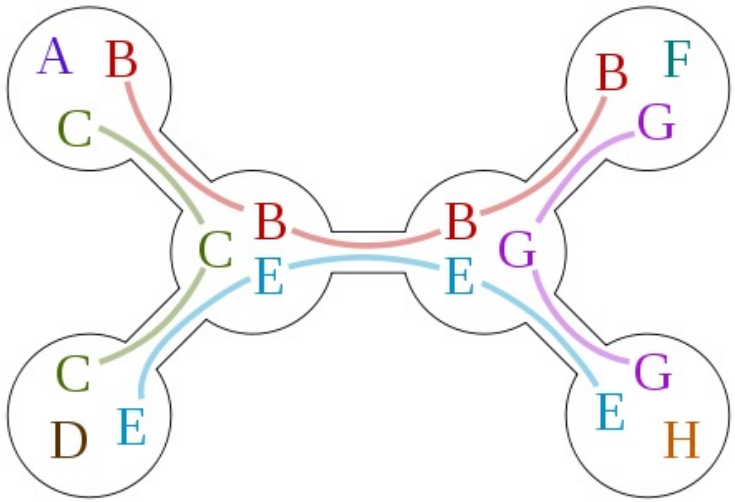


Tree-width of non-oriented graph

Tree-decomposition of a graph:

1. Each graph vertex is associated with at least one tree node.
2. Each graph edge connects two vertices that are listed together at some tree node.
3. Each graph vertex is listed at the nodes of a contiguous sub-tree of the tree.



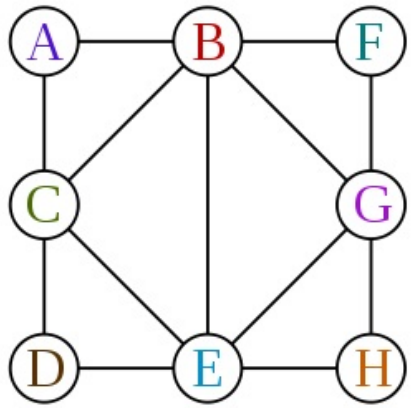


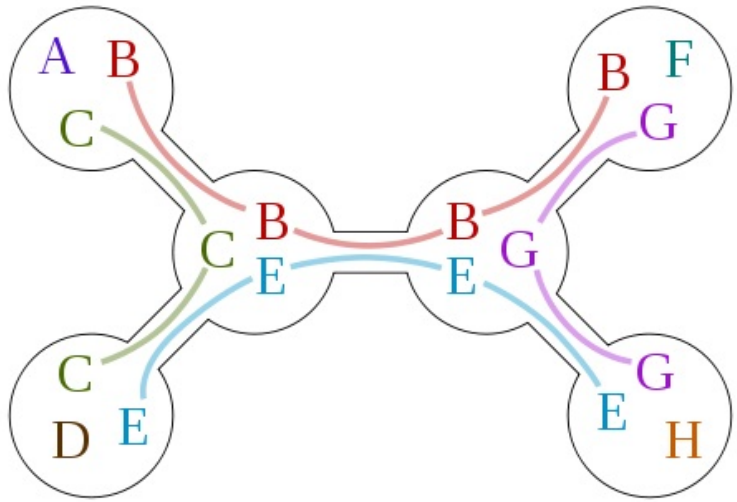
Tree-width of non-oriented graph

Tree-decomposition of a graph:

1. Each graph vertex is associated with at least one tree node.
2. Each graph edge connects two vertices that are listed together at some tree node.
3. Each graph vertex is listed at the nodes of a contiguous sub-tree of the tree.

The width of a tree-decomposition is the size of its largest node minus one.



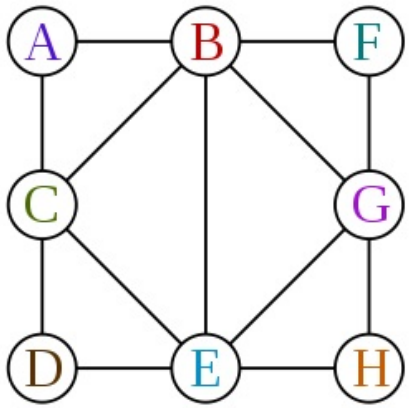


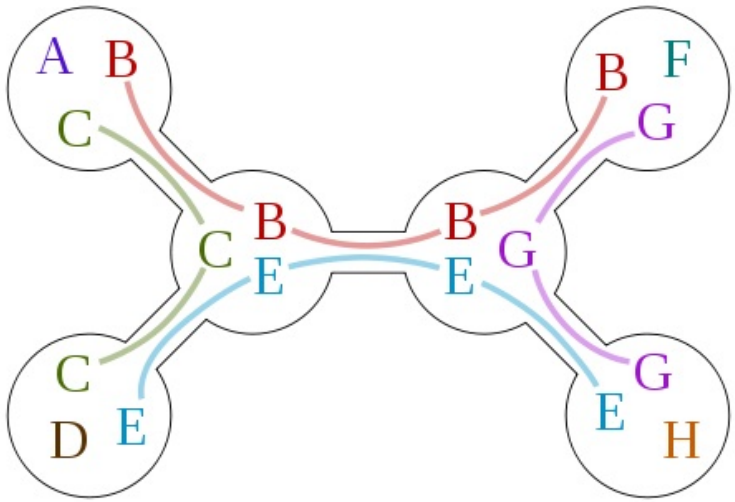
Tree-width of non-oriented graph

Tree-decomposition of a graph:

1. Each graph vertex is associated with at least one tree node.
2. Each graph edge connects two vertices that are listed together at some tree node.
3. Each graph vertex is listed at the nodes of a contiguous sub-tree of the tree.

The width of a tree-decomposition is the size of its largest node minus one.
The tree-width of a graph is the minimum width among all its possible tree-decompositions.





Bounded tree-widths

Theorem. Deciding the existence of a minimal siphon containing a given set of places Q can be done in linear time for Petri-nets of bounded tree-width.

Proof. Definable in Monadic Second Order logic $\Longrightarrow$ Recognizable in linear time.

1990 *Definable monadic second-order logic of graphs I. Recognizable sets of Finite Graphs.* Bruno Courcelle. Information and Computation.

Bounded tree-widths

Theorem. Deciding the existence of a minimal siphon containing a given set of places Q can be done in linear time for Petri-nets of bounded tree-width.

Proof. Definable in Monadic Second Order logic $\Longrightarrow$ Recognizable in linear time.

1990 *Definable monadic second-order logic of graphs I. Recognizable sets of Finite Graphs.* Bruno Courcelle. Information and Computation.

$$\textsc{siphon}(\mathrm{S}): \quad \forall v(S(v) \Rightarrow \mathsf{place}(v)) \wedge \exists v(S(v)) \wedge \forall t(\exists v(S(v) \wedge \mathsf{edge}(t,v)) \Rightarrow \exists v(S(v) \wedge \mathsf{edge}(v,t)))$$

Bounded tree-widths

Theorem. Deciding the existence of a minimal siphon containing a given set of places Q can be done in linear time for Petri-nets of bounded tree-width.

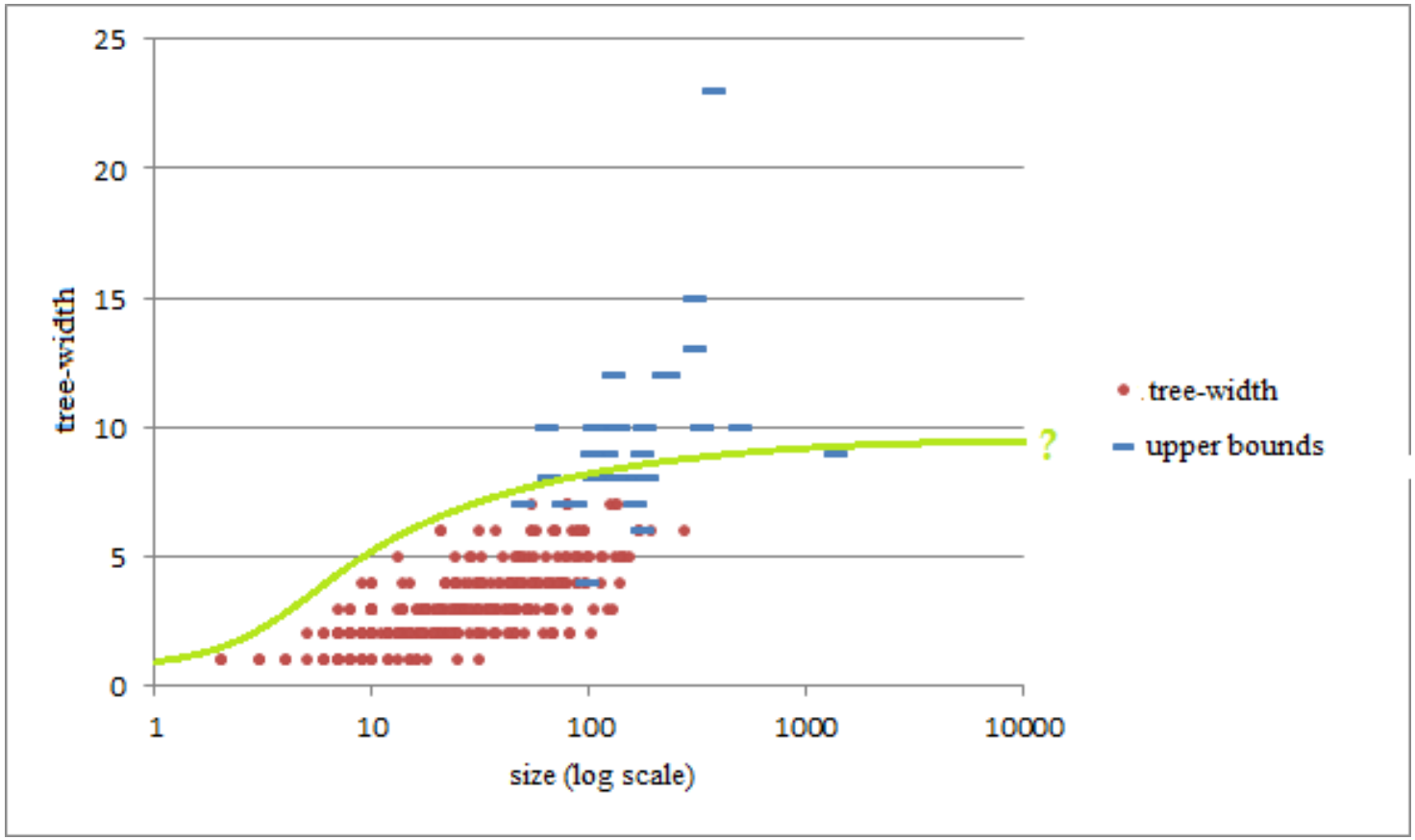
Proof. Definable in Monadic Second Order logic $\Longrightarrow$ Recognizable in linear time.

> 1990 *Definable monadic second-order logic of graphs I. Recognizable sets of Finite Graphs.* Bruno Courcelle. Information and Computation.

$$\textsc{siphon}(\mathrm{S}): \quad \forall v(S(v) \Rightarrow \mathsf{place}(v)) \wedge \exists v(S(v)) \wedge \forall t(\exists v(S(v) \wedge \mathsf{edge}(t,v)) \Rightarrow \exists v(S(v) \wedge \mathsf{edge}(v,t)))$$

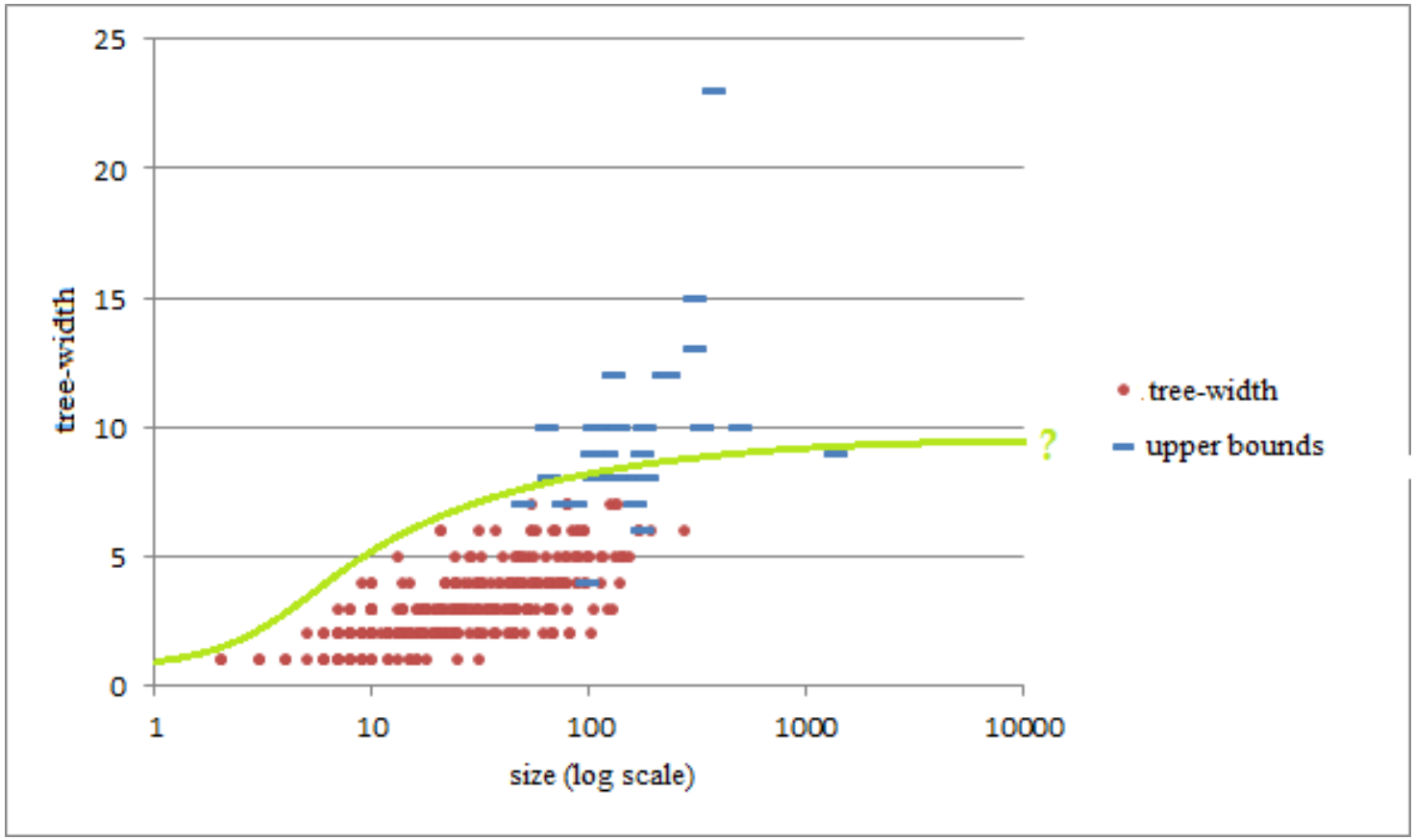
$$\textsc{Min-Siphon}(\mathrm{Q}): \quad \exists S(\textsc{siphon}(S) \wedge \forall S'(\textsc{siphon}(S') \wedge \forall v(S'(v) \Rightarrow S(v)) \Rightarrow \forall v(S(v) \Rightarrow S'(v))) \wedge \forall v(Q(v) \Rightarrow v \in S))$$

Biochemicals networks seem to have a **bounded** tree-width



Biomodels.net tree-width as a function of the size (places and transitions) of the Petri net Computed tree-width ≤ 10

Biochemicals networks seem to have a **bounded** tree-width



Biomodels.net tree-width as a function of the size (places and transitions) of the Petri net Computed tree-width ≤ 10

Does not explain our good performance but suggests that the problem is tractable.

Siphon-Trap Property (STP) in MSO

Theorem. Deciding the STP in linear time for Petri nets of bounded tree-width.

Siphon-Trap Property (STP) in MSO

Theorem. Deciding the STP in linear time for Petri nets of bounded tree-width.

Proof. $\textsc{trap}(\mathrm{S})$:

$$\begin{aligned}&\forall v(S(v) \Rightarrow \mathsf{place}(S))\\ &\wedge \exists v(S(v))\\ \wedge \forall t(\exists v(S(v) \wedge \mathsf{edge}(v,t)) &\Rightarrow \exists v(S(v) \wedge \mathsf{edge}(t,v)))\end{aligned}$$

$\textsc{stp}$:

$$\begin{aligned}&\forall S(\textsc{siphon}(S) \Rightarrow\\ &\exists S'(\forall v(S'(v) \Rightarrow S(v)) \wedge \textsc{trap}(S')\\ &\wedge \exists v(S'(v) \wedge \mathsf{marked}(v))))\end{aligned}$$

Finite domain model for T-/P-invariants

Minimal T-invariants

Dynamical characterisation: Multi-set of transitions whose weighted firing produces any initial marking.

Minimal T-invariants

Dynamical characterisation: Multi-set of transitions whose weighted firing produces any initial marking.

Structural characterisation: A vector V of integers solution of $I.V = 0$. I is the incidence (stoichiometric) matrix.
$I = Post - Pre$

Minimal T-invariants

Dynamical characterisation: Multi-set of transitions whose weighted firing produces any initial marking.

Structural characterisation: A vector V of integers solution of $I.V = 0$. I is the incidence (stoichiometric) matrix.
$I = Post - Pre$

A T-invariant is minimal if its support is not the support of any other T-invariant, and the greatest common divisor of all entries is 1.

CSP model for P/T-invariants

A Petri net with n places and m transitions

variables

m Finite Domains variables.

constraints

n (linear) equality constraints

$$\forall 1 \leq i \leq n,\ Pre_i \cdot V = Post_i \cdot V$$

Finding (some) T-invariants is reduced to finding integer assignments satisfying these formulas.

Enumerating all minimal T-invariants

CSP model for an invariant V:

$$\forall 1 \leq i \leq n,\ Pre_i \cdot V = Post_i \cdot V \wedge V \cdot \mathbf{1} > 0$$

repeat

find a solution, enumerating from low to high

add the solution to the basis

remove non-minimal T-invariants from the basis if there are any

post the new constraint

$$\forall B \in \mathcal{B} \prod_{B_i \neq 0} V_i = 0$$

until no solution found

Experimental results: minimal T-invariants enumeration

Using GNU-Prolog solver:

404 models of Biomodels.net

- 6 models include non-integer stoichiometry
- 22 models could not be solved within 10 min
- 376 models: all minimal T-invariants of each model are enumerated in less than 10 seconds

80 models of Petriweb

- all instances in less than 1 second

Experimental results: minimal P-invariants enumeration

Using GNU-Prolog solver:

404 **models of Biomodels.net**

- all instances in less than 1 second

80 **models of Petriweb**

- all instances in less than 1 second

T-invariants and Steady states

$R = \{R_1, R_2, ..., R_m\}$ reactions, $X = \{X_1, X_2, .., X_n\}$ species.

$$R_j\colon \sum_{i=1}^{n} Pre_{ij} X_i \stackrel{e_j}{\Rightarrow} \sum_{i=1}^{n} Post_{ij} X_i \qquad Pre_{ij} \text{ and } Post_{ij} \text{ positive integers}$$

T-invariants and Steady states

$R = \{R_1, R_2, ..., R_m\}$ reactions, $X = \{X_1, X_2, .., X_n\}$ species.

R_j: $\sum_{i=1}^{n} Pre_{ij} X_i \stackrel{e_j}{\Rightarrow} \sum_{i=1}^{n} Post_{ij} X_i$ Pre_{ij} and $Post_{ij}$ positive integers

Finding a steady state amounts to solving:

$$\forall X_i \in X, \frac{dX_i}{dt} = 0$$

equivalent to:

$$I \cdot E(X) = 0 \text{ where } E = (e_1, \ldots, e_m).$$

T-invariants and Steady states

$R = \{R_1, R_2, ..., R_m\}$ reactions, $X = \{X_1, X_2, .., X_n\}$ species.

R_j: $\sum_{i=1}^{n} Pre_{ij} X_i \stackrel{e_j}{\Rightarrow} \sum_{i=1}^{n} Post_{ij} X_i$ Pre_{ij} and $Post_{ij}$ positive integers

Finding a steady state amounts to solving:

$$\forall X_i \in X, \frac{dX_i}{dt} = 0$$

equivalent to:

$$I \cdot E(X) = 0 \text{ where } E = (e_1, \ldots, e_m).$$

➡Let $V = \sum \alpha_j V'_j$. (V'_j is a minimal T-invariants, $\alpha_j \in \mathbb{R}^+$) Solving

$E(X) = V$ will lead to steady states of the original system.

Solving E(X)=V for computing steady states

Given (V) a linear combination of minimal T-invariants:

For support:

Restriction to General Mass Action

➟Gauss Elimination on log-linearised equations

For null entries of V:

Restriction to multiplicative kinetics

$e_j(X) = 0 \Leftrightarrow \exists i, Pre_{ij} > 0 \wedge X_i = 0$

➟simple enumeration

Solving E(X)=V for computing steady states

Given (V) a linear combination of minimal T-invariants:

For support:

Restriction to General Mass Action
➟Gauss Elimination on
log-linearised equations

For null entries of V:

Restriction to multiplicative kinetics
$e_j(X) = 0 \Leftrightarrow \exists i, Pre_{ij} > 0 \wedge X_i = 0$
➟simple enumeration

Solve (1) to compute steady states:

$$(1) \begin{cases} (a) & k_j \prod_{i=1}^{n} X_i^{Pre_{ij}} = V_j & j \in support(V) \\ (b) & \exists i, Pre_{ij} > 0 \wedge X_i = 0 & j \notin support(V) \end{cases}$$

Solving E(X)=V for computing steady states

Given (V) a linear combination of minimal T-invariants:

For support:

Restriction to General Mass Action
➟Gauss Elimination on
log-linearised equations

For null entries of V:

Restriction to multiplicative kinetics
$e_j(X) = 0 \Leftrightarrow \exists i, Pre_{ij} > 0 \wedge X_i = 0$
➟simple enumeration

Solve (1) to compute steady states:

$$(1) \begin{cases} (a) & k_j \prod_{i=1}^{n} X_i^{Pre_{ij}} = V_j \qquad j \in support(V) \\ (b) & \exists i, Pre_{ij} > 0 \wedge X_i = 0 \quad j \notin support(V) \end{cases}$$

2000 *Biochemical systems analysis of genome-wide expression data.*
E. O. Voit and T. Radivoyevitch. Bioinformatics.

Heuristic enumeration of steady states

Infinity of T-invariants $\Rightarrow$ Restriction to specific combinations of minimal invariants.

$$(1) \begin{cases} (a) & k_j \prod_{i=1}^{n} X_i^{Pre_{ij}} = V_j \qquad j \in support(V) \\ (b) & \exists i, Pre_{ij} > 0, X_i = 0 \quad j \notin support(V) \end{cases}$$

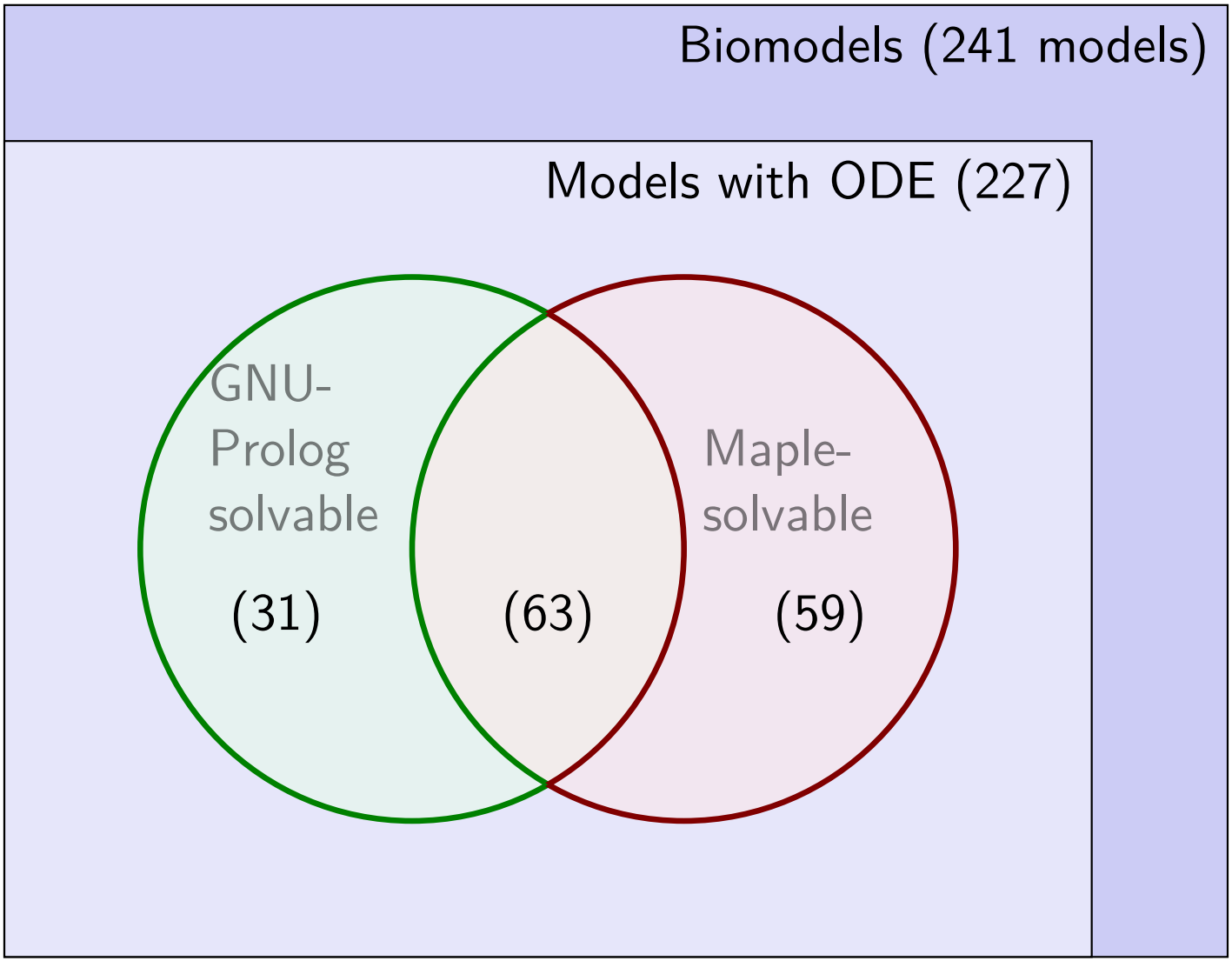
Look for minimal T-invariants;
For minimal T-invariant combinations (starting with 0 or 1 invariant), try to solve the system (1)
if (b) is not satisfiable **then**
  add another minimal T-invariant (heuristic)
else
  try to add all other minimal T-invariants and stop if (a) is unsatisfiable
end if

Other choices

- We can apply this approach to any T-invariant (even not minimal).
- The method does not need all minimal T-invariants.
- The computational cost of trying to compute steady states is low, compared to the T-invariant computation.

Steady states computation

Analytical solutions computed in less than 2 min:



Conclusion

- Petri nets structural properties can give us some information about the biochemical network dynamics when the kinetics are missing.

Conclusion

- Petri nets structural properties can give us some information about the biochemical network dynamics when the kinetics are missing.
- Constraint programming over finite domain can be successfully applied to Petri nets structural problems.
- GNU-Prolog and miniSAT solvers outperform state-of-the-art algorithms for enumerating minimal siphons.

Conclusion

- Petri nets structural properties can give us some information about the biochemical network dynamics when the kinetics are missing.
- Constraint programming over finite domain can be successfully applied to Petri nets structural problems.
- GNU-Prolog and miniSAT solvers outperform state-of-the-art algorithms for enumerating minimal siphons.
- Surprisingly good performance on real-size practical models.
- Linear-time complexity result for Petri nets with bounded tree-width.

Perspectives

- Add biological constraints to the computation of T-invariants (compute only those of interest).

Perspectives

- Add biological constraints to the computation of T-invariants (compute only those of interest).
- Provide a generic tool for the verification of graphs structural properties having a biological meaning, with an underlying CLP engine. (e.g. model reduction as sub-graph epimorphism)

Perspectives

- Add biological constraints to the computation of T-invariants (compute only those of interest).
- Provide a generic tool for the verification of graphs structural properties having a biological meaning, with an underlying CLP engine. (e.g. model reduction as sub-graph epimorphism)
- Identify parameters for certain structural properties computation that ensure no performance issues.
- Understand the links between structural properties of graphs and the practical complexity of solving NP-hard problems on them.