Crocodile: a Symbolic/Symbolic tool for the analysis of Symmetric Nets with Bags

M. Colange

LIP6, CNRS UMR 7606, Université P. & M. Curie – Paris 6

MeFoSyLoMa
23 mars 2012

Context

Tackle the combinatorial explosion of the state space :

- symmetries $\longrightarrow$ quotient graph
- decision diagrams

Crocodile: symbolic-symbolic:

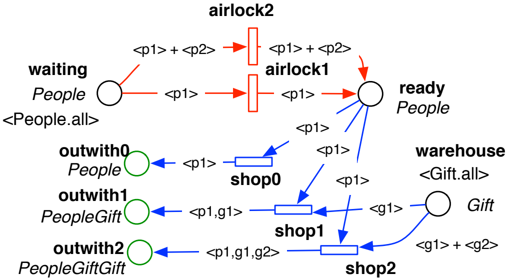
- Symmetric Nets with Bags (SNB) [Haddad et al., 2009]
  symbolic reachability graph (SRG)
- Hierarchical Set Decision Diagrams (SDD)
  [Couvreur and Thierry-Mieg, 2005]
  symbolic set manipulation

Symbolic-Symbolic: history

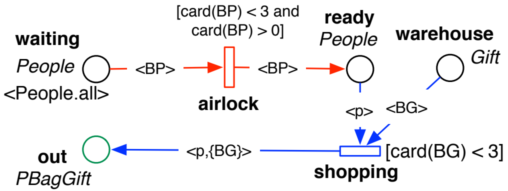
- [Clarke et al., 1996] mitigated results
  - Binary Decision Diagrams
  - operations over BDD encoded as BDD
- [Thierry-Mieg et al., 2004] first interesting experiments
- [Colange et al., 2011] Crocodile first application (to SNB)
  - Hierarchical DD
  - operations encoded as homomorphisms

SN vs. SNB

Class
People is 1..P;
Gift is 1..G;
Domain
PeopleGift is <People, Gift>;
PeopleGiftGift is <People, Gift, Gift>;
Var
p1, p2 in People;
g1, g2 in Gift;



Domain
BagPeople is Bag(People);
BagGift is Bag(Gift);
PBagGift is <People, BagPeople>;
Var
p in People;
BP in BagPeople;
BG in Bag(Gift);



Symbolic Reachability Graph (SRG)

σ, permutation of the states and transitions of the system, is a symmetry iff: if $s_1 \mapsto_t s_2$, then $\sigma.s_1 \mapsto_{\sigma.t} \sigma.s_2$.

- the set of system symmetries is a group
- SRG $=$ RG$_{/G}$ (quotient graph)
- SRG preserves accessibility and (symmetric) CTL* properties

Symmetries in SNB : symbolic markings

Two concrete markings for the place "out"

- $M_1 : M(out) =< p_1, \{g_1, g_2\} > + < p_2, \{g_1, g_2\} >$
- $M_2 : M(out) =< p_2, \{g_1, g_2\} > + < p_3, \{g_1, g_2\} >$

$\rightarrow$ differ only by a permutation of constants.

A symbolic marking encompasses them [Chiola et al., 1993]

- $M(out) =< P_0, \{R_0\} >$ with $|P_0| = 2$ and $|R_0| = 2$

Symmetries in SNB : symbolic markings

Two concrete markings for the place "out"

- $M_1 : M(out) =< p_1, \{g_1, g_2\} > + < p_2, \{g_1, g_2\} >$
- $M_2 : M(out) =< p_2, \{g_1, g_2\} > + < p_3, \{g_1, g_2\} >$

$\rightarrow$ differ only by a permutation of constants.

A symbolic marking encompasses them [Chiola et al., 1993]

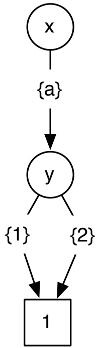
- $M(out) =< P_0, \{R_0\} >$ with $|P_0| = 2$ and $|R_0| = 2$
  - What if R_0 is the marking of another place ?

SDD

- Set DD : set assignments
- edges labels = sets
- Hierarchical : edges labels = SDD

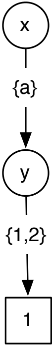
SDD

sequences of assignments : $x = a,\ y = 1$ and $x = a,\ y = 2$



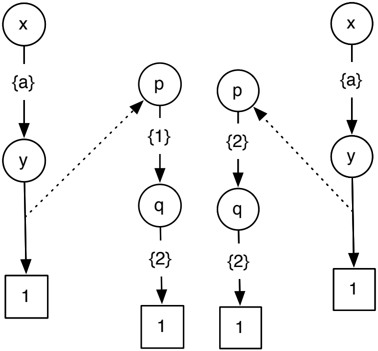
SDD

sequences of assignments : $x = a,\ y = 1$ and $x = a,\ y = 2$



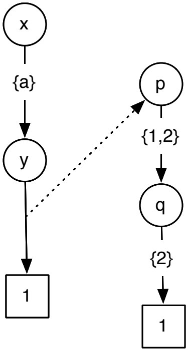
SDD

sequences of assignments : $x = a$, $y = 1$ and $x = a$, $y = 2$
and now y is a struct : $y = \{p : 1, q : 2\}$, $y = \{p : 2, q : 2\}$



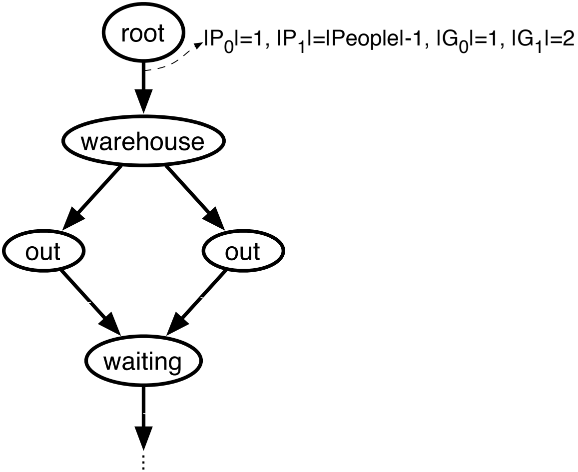
SDD

sequences of assignments : $x = a$, $y = 1$ and $x = a$, $y = 2$
and now y is a struct : $y = \{p : 1, q : 2\}$, $y = \{p : 2, q : 2\}$



Encoding a symbolic marking with a SDD

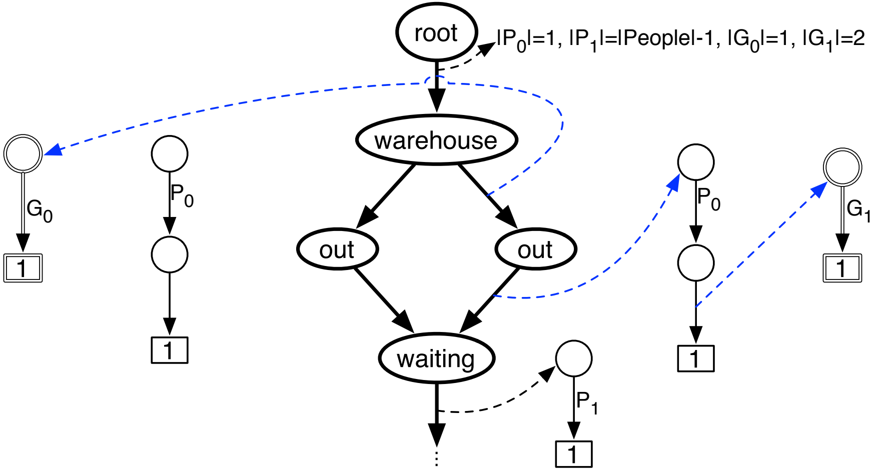
M(warehouse) $= < G_0 >$, M(out) $= < P_0, \{G_1\} >$, M(waiting) $= < P_1 >$
M(warehouse) $= < G_1 >$, M(out) $= < P_0, \{G_0\} >$, M(waiting) $= < P_1 >$



Encoding a symbolic marking with a SDD

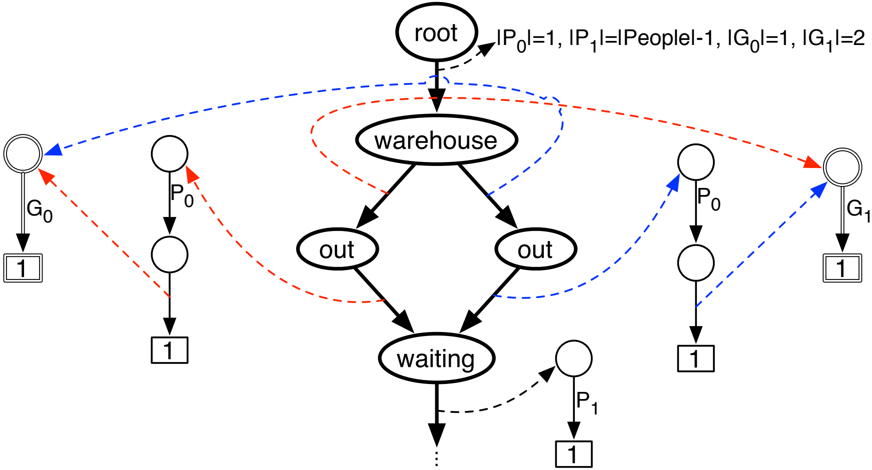
M(warehouse) $= <G_0>$, M(out) $= <P_0, \{G_1\}>$, M(waiting) $= <P_1>$

M(warehouse) $= <G_1>$, M(out) $= <P_0, \{G_0\}>$, M(waiting) $= <P_1>$



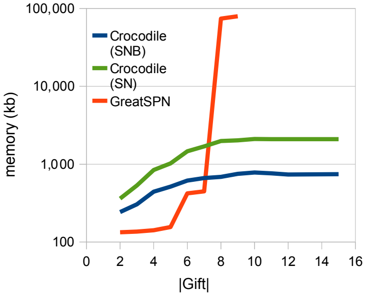
Encoding a symbolic marking with a SDD

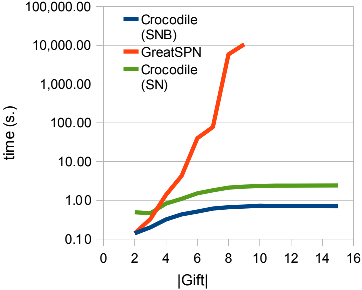
M(warehouse) = < G_0 >, M(out) = < $P_0, \{G_1\}$ >, M(waiting) = < P_1 >
M(warehouse) = < G_1 >, M(out) = < $P_0, \{G_0\}$ >, M(waiting) = < P_1 >



Towards Crocodile 2

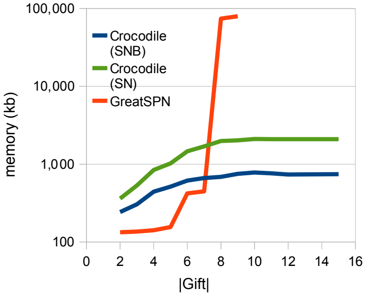
[Colange et al., 2011]

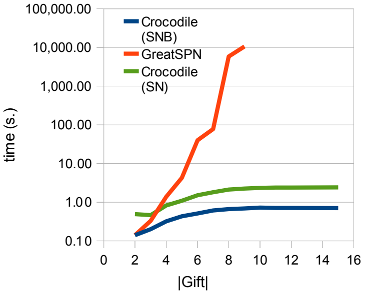




Towards Crocodile 2

[Colange et al., 2011]

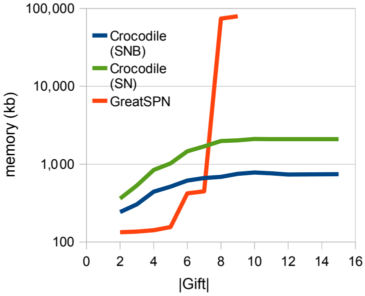


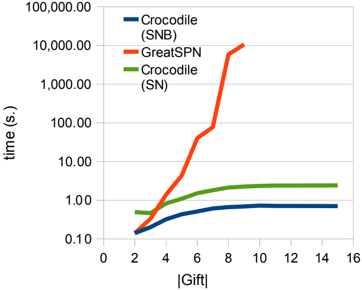


- demonstrate the feasibility of the symbolic-symbolic approach

Towards Crocodile 2

[Colange et al., 2011]

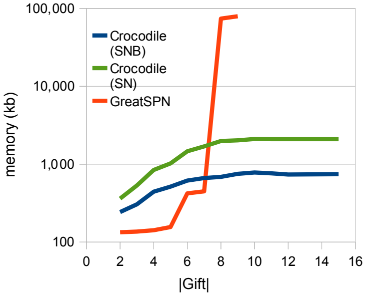


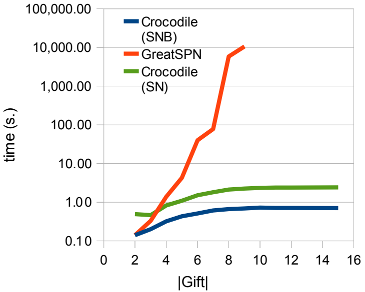


- demonstrate the feasibility of the symbolic-symbolic approach
- demonstrate efficiency of bags

Towards Crocodile 2

[Colange et al., 2011]





- demonstrate the feasibility of the symbolic-symbolic approach
- demonstrate efficiency of bags
- performance is hindered by a poor handling of bags

Recursive representation of sets of bags

Bounds over cardinal of bags are known

- recursion over cardinality
  - $Bag_n(C) = \{B \in Bag(C) | card(B) = n\}$
  - $Bag_n(C) \uplus Bag_p(C) = Bag_{n+p}(C)$

Thanks to Pierre Parutto

Recursive representation of sets of bags

Bounds over cardinal of bags are known

- recursion over cardinality
  - $Bag_n(C) = \{B \in Bag(C) | card(B) = n\}$
  - $Bag_n(C) \uplus Bag_p(C) = Bag_{n+p}(C)$
- recursion over the support (static equivalence classes of the values)
  - $Bag_n(C_1 \cup \ldots C_k) = (\cup_{i=1}^{k} Bag_n(C_i)) \cup Bag_n^*(C_1, \ldots C_k)$
  - $Bag_n^*(C_1, \ldots C_k) =$ bags with "mixed" support

Thanks to Pierre Parutto

Recursive representation of sets of bags

Bounds over cardinal of bags are known

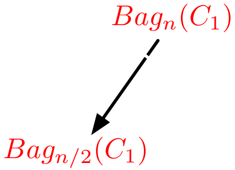
- recursion over cardinality
  - $Bag_n(C) = \{B \in Bag(C) | card(B) = n\}$
  - $Bag_n(C) \uplus Bag_p(C) = Bag_{n+p}(C)$
- recursion over the support (static equivalence classes of the values)
  - $Bag_n(C_1 \cup \ldots C_k) = (\cup_{i=1}^{k} Bag_n(C_i)) \cup Bag_n^*(C_1, \ldots C_k)$
  - $Bag_n^*(C_1, \ldots C_k) =$ bags with "mixed" support
- Anonymization: values in types are contextual

Thanks to Pierre Parutto

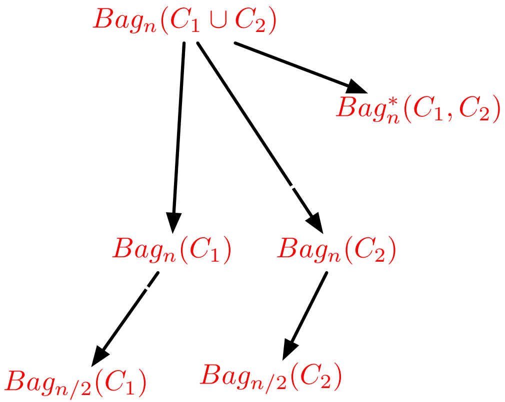
Crocodile2: new bag representation

$$Bag_n(C_1)$$

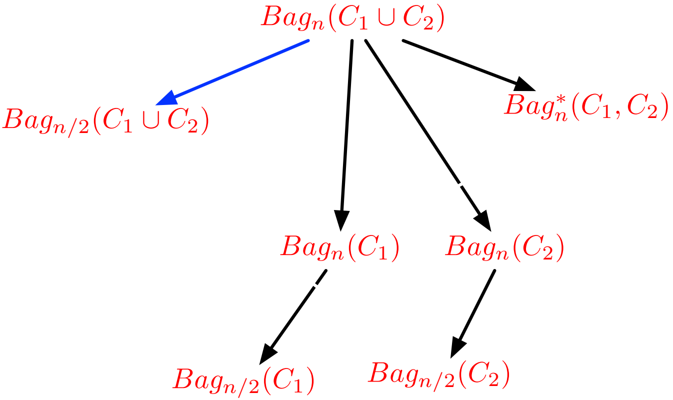
Crocodile2: new bag representation



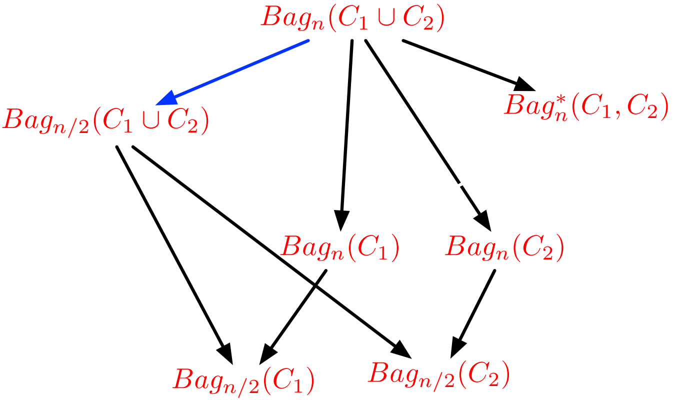
Crocodile2: new bag representation



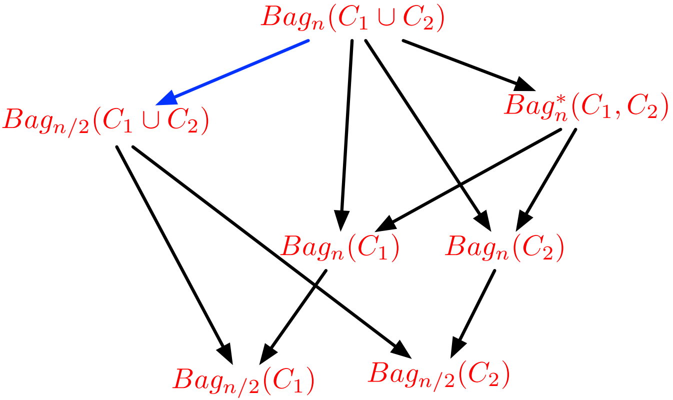
Crocodile2: new bag representation



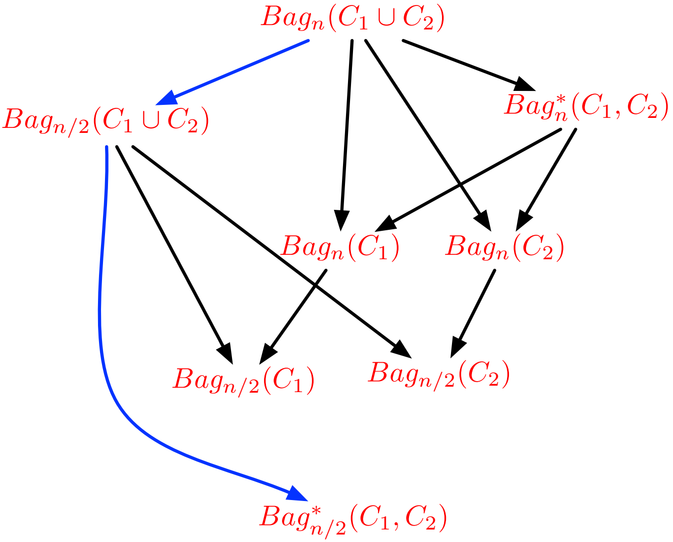
Crocodile2: new bag representation



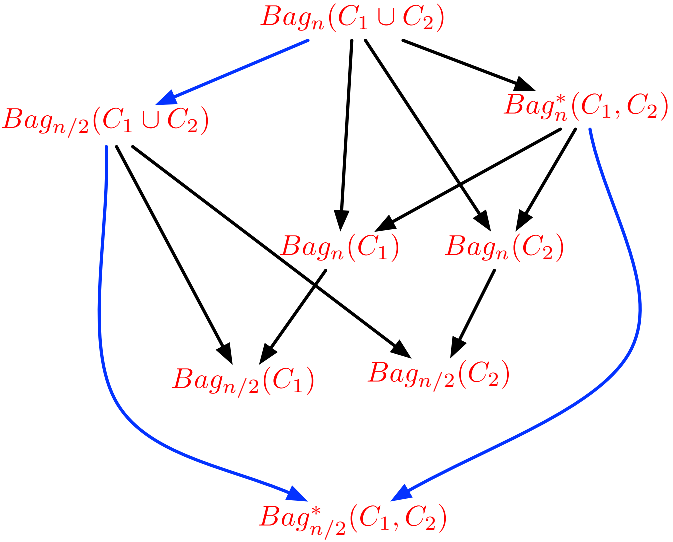
Crocodile2: new bag representation



Crocodile2: new bag representation



Crocodile2: new bag representation



CosyVerif

- Crocodile1: integrated in Alligator/CosyVerif
- Crocodile2: finalization, integrated ASAP

CosyVerif

- online shared tools integration platform
- LIPN + LIP6 + LSV + MeFoSyLoMa
- more details this afternoon

Future

- Model Checking Contest 2012
- generalization of the symbolic-symbolic approach
    - promising results with no hierarchy [Baarir et al., 2012]
    - symbolic-symbolic + hierarchy

Baarir, S., Colange, M., Kordon, F., and Thierry-Mieg, Y. (2012).
State space analysis using symmetries on decision diagrams.
Application of Concurrency to System Design (to appear).

Chiola, G., Dutheillet, C., Franceschinis, G., and Haddad, S. (1993).
Stochastic well-formed colored nets and symmetric modeling applications.
Computers, IEEE Transactions on, 42(11):1343–1360.

Clarke, E., Enders, R., Filkorn, T., and Jha, S. (1996).
Exploiting symmetry in temporal logic model checking.
Formal Methods in System Design, 9(1):77–104.

Colange, M., Baarir, S., Kordon, F., and Thierry-Mieg, Y. (2011).
Crocodile: a symbolic/symbolic tool for the analysis of symmetric nets with bag.
Applications and Theory of Petri Nets, pages 338–347.

Couvreur, J. and Thierry-Mieg, Y. (2005).
Hierarchical decision diagrams to exploit model structure.
Formal Techniques for Networked and Distributed Systems-FORTE 2005, pages 443–457.

Haddad, S., Kordon, F., Petrucci, L., Pradat-Peyre, J., and Treves, L. (2009).
Efficient state-based analysis by introducing bags in petri nets color domains.
In *American Control Conference, 2009. ACC'09.*, pages 5018–5025. IEEE.

Thierry-Mieg, Y., Ilié, J., and Poitrenaud, D. (2004).
A symbolic symbolic state space representation.
Formal Techniques for Networked and Distributed Systems–FORTE 2004, pages 276–291.