A Global Constraint for Closed Frequent Itemset Mining

N. Lazaar[2] Y. Lebbah[1] S. Loudni[3] **M. Maamar**[1,2]
V. Lemière[3] C. Bessiere[2] P. Boizumault[3]

[1] Lab. LITIO, University of Oran 1 – Algeria
[2] Lab. LIRMM, University of Montpellier – France
[3] Lab. GREYC, University of Caen – France

CP 2016

Toulouse, *France*, 6 *September* 2016

Itemset Mining: Definition

Data mining
- $\boxed{\text{Itemset}}$
- Sequence
- Graph
- …

- It aims at finding regularities in a dataset

- *Find sets of products that are frequently bought together*

Itemset Mining: Example

trans	items							
t1		B	C				G	H
t2	A			D				
t3	A		C	D				H
t4	A				E	F		
t5		B			E	F	G	

Set of items: I = {A,B,C,D,E,F,G,H}

Set of transaction: T = {t1,t2,t3,t4,t5}

Itemset: $P \subseteq I$

Cover:

Cover(AD)= {t2,t3}

Cover(BEFG)= {t5}

Closed Frequent Itemset Mining

- The frequency of a itemset is the size of its cover.
- $\theta \in \mathbf{N}^+$ be the minimum support.

Frequent itemset mining problem :

- Extract all itemsets P satisfying : freq(P) ≥ θ

trans	items							
t1		B	C		E		G	H
t2	A			D	E	F		
t3	A		C	D				H
t4	A			D	E	F		
t5		B			E	F	G	

freq(BC) = 1

Frequent itemset with θ = 3

freq(EG) = 2

A:3
D:3
E:4
F:3
AD:3
EF:3

are not closed

Can we compact these itemsets ?

Closed itemsets

are closed

Closed Frequent Itemset Mining

The Need:

Extract **all** Closed Frequent Itemsets

How ?

Closed Frequent Itemset: State of art

→ **Many efficient algorithms for mining closed itemsets** ✓

but

- dedicated to particular classes of constraints

  ↳ adding new constraints requires new implementations.

→ **CP framework:**

- **modeling**

  - in a declarative way -> facilitates the addition of new constraints

- **solving**

  - efficient solvers based on filtering

State of the art: CP approach

- **Reified model** **[De Raedt et al,2008]:**

Variables:

item variables: decision

transaction variables: auxiliary

Reified Constraints:

Coverage

Frequency

Closedness

Drawbacks

- Additional dimension of transaction variables.
- The huge number of reified constraints -> limitation

Contribution

Proposition: a **global constraint** that encodes efficiently the Closed Frequent itemsets Mining problem.

- Domain consistency with polynomial algorithm.
- No reified constraints/extra variables.
- Backtrack-free.

ClosedPattern: Definition

- Encoding: $P = \{P_1 \ldots P_n\}$: $D(P_i) = \{0,1\}$
- Definition:

$$\textsc{ClosedPattern}_{D,\theta}(P): (\text{freq}(P) \geq \theta) \land (P \text{ is a closed pattern})$$

trans	items							
t1		B	C		E		G	H
t2	A			D	E			
t3	A		C	D				H
t4	A			D	E	F		
t5		B			E	F	G	

$$\textsc{ClosedPattern}_{D,2}(\text{AD}) \begin{cases} \text{freq(AD)} > 2 \\ \text{AD is closed} \end{cases}$$

CLOSEDPATTERN: Filtering rules

3 Filtering rules

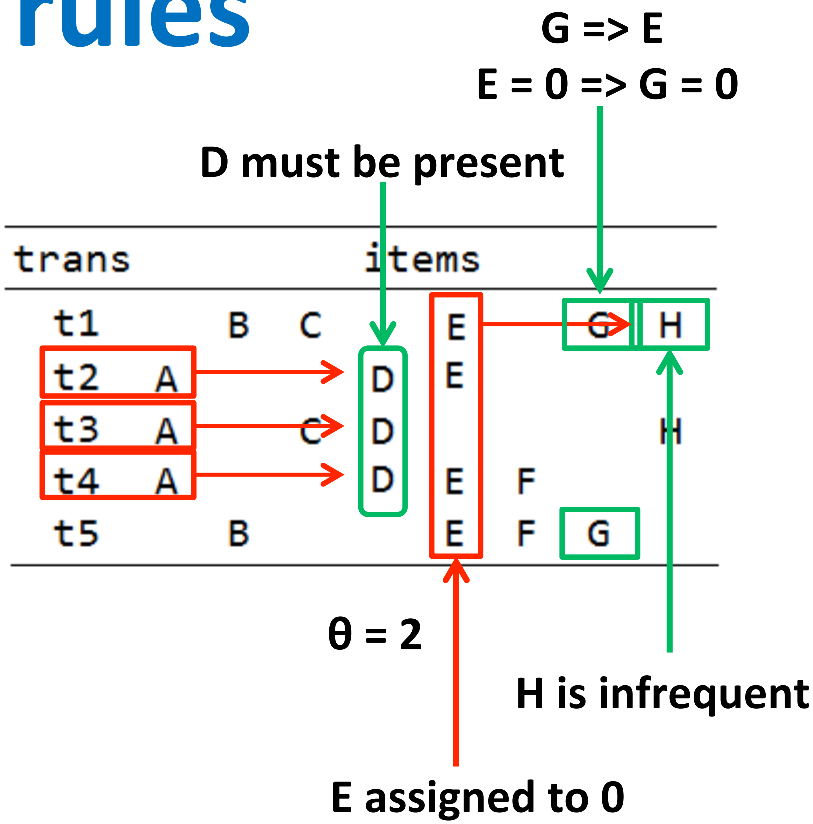
$0 \notin D(P_j)$:

Rule 1: full extension items

$1 \notin D(P_j)$:

Rule 2: infrequent items

Rule 3: Absent items



Filtering algorithm & Complexity

Algorithm: n: items, m: transactions

for each free variable

- **rule 1:** maintained ($O(n \times m)$)
- **rule 2:** maintained ($O(n \times m)$)

for each absent item

- **rule 3:** maintained ($O(n \times (n \times m))$)

Time: $O(n \times (n \times m))$: Cubic

Space: $O(n \times m)$: Quadratic

ClosedPattern vs Reified

trans	A	B	C	D	E	F	G	H
t1		B	C				G	H
t2	A			D				
t3	A		C	D				H
t4	A				E	F		
t5		B			E	F	G	

$\theta = 2$

Lex : variables

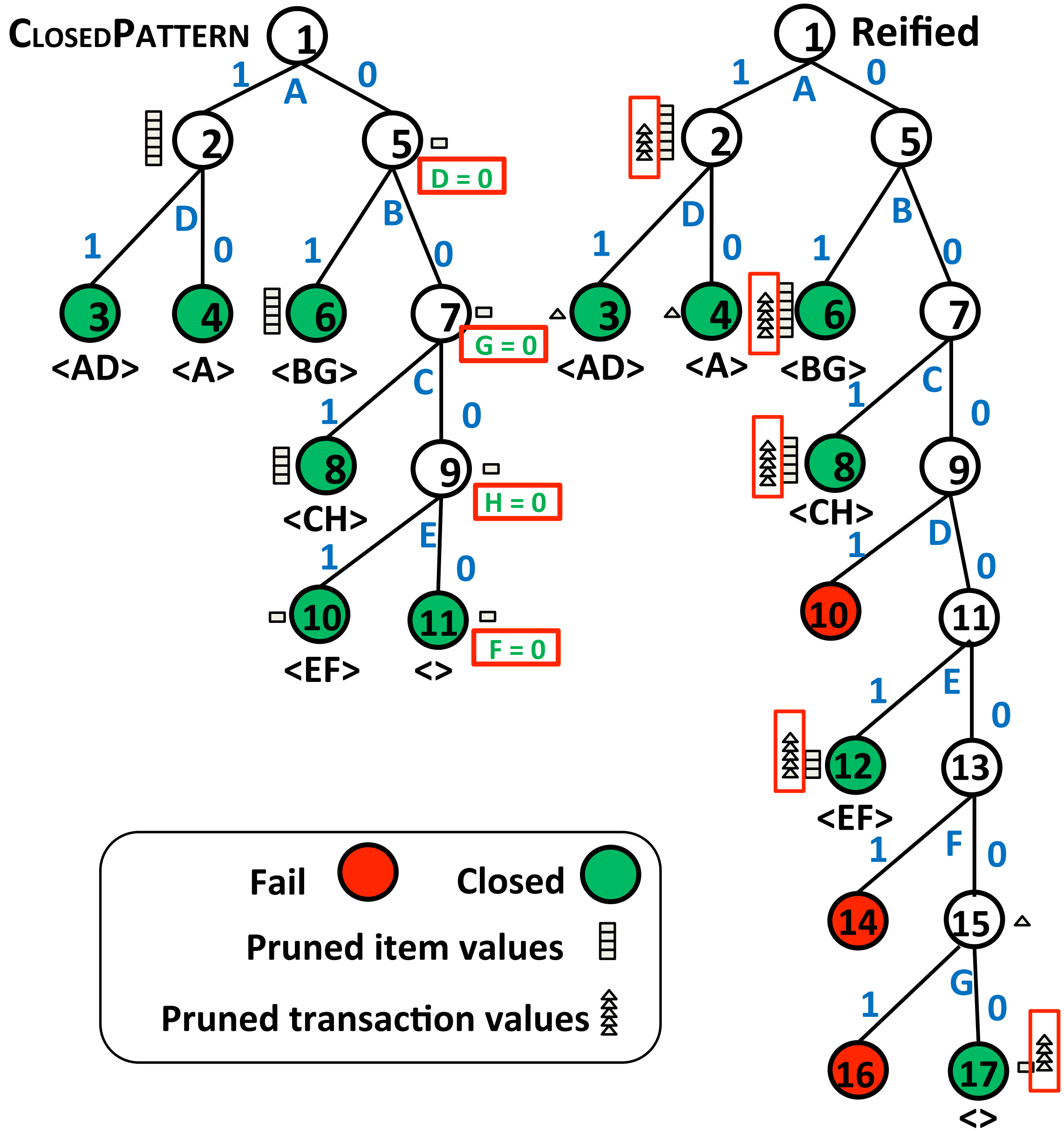
max_val : values

At each node:

$0 \notin D(P_j)$: rule1

$1 \notin D(P_j)$: rule2

$\boxed{1 \notin D(P_j) : \text{rule3}}$



ClosedPattern: Backtrack-free

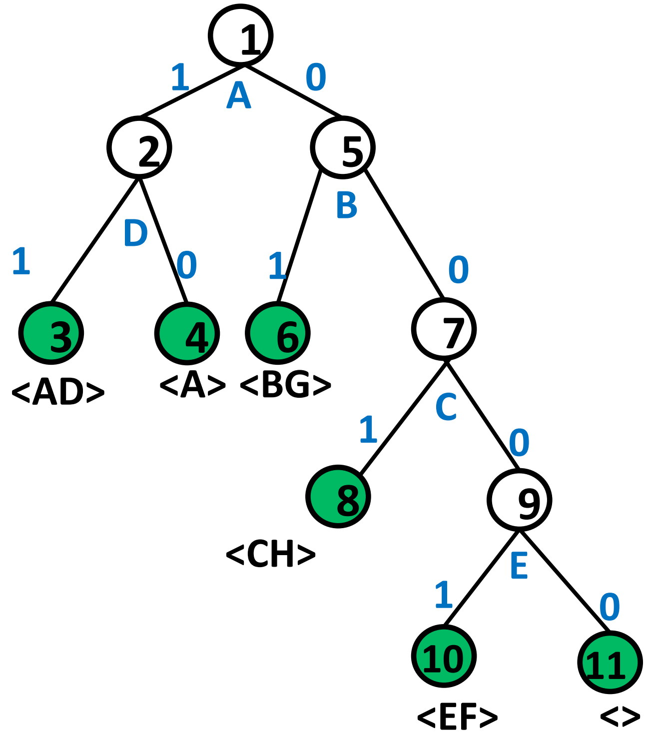
- DC at *each node* on *Boolean variables*.

- **Tree search size:**

full binary tree -> $2 \times |C| - 1$

- **Backtrack-free:**

$O(|C| \times n^2 \times m)$



ClosedPattern: Experiments

Dataset	$\vert\mathcal{T}\vert$	$\vert\mathcal{I}\vert$	ρ
Chess	3 196	75	49%
Splice1	3 190	287	21%
Mushroom	8 124	119	19%
Connect	67 557	129	33%
BMS-Web-View1	59 602	497	0.5%
T10I4D100K	100 000	1 000	1%
T40I10D100K	100 000	1 000	4%
Pumsb	49 046	7 117	1%
Retail	88 162	16 470	0.06%

- **Comparison with:**
  - The most efficient CP method: CP4IM (reified)
  - The most efficient ad hoc algorithm: LCM-v5.2

- **Solver**: or-tools, Intel Xeon E5-2680@ 2,5 GHz with 128 Gb

- **ClosedPattern-DC:** rules 1,2 and 3 (cubic pruning)
- **ClosedPattern-WC:** rules 1 and 2 (quadratic pruning)

ClosedPattern: Experiments

ClosedPattern-DC vs CP4IM:

Dataset	θ	Times(s)		#Failures	
	%	DC	CP4IM	DC	CP4IM
Chess	30	**45.92**	136.31	**0**	1054
	20	**187.89**	467.52	**0**	32 381
	10	**969.40**	1950.5	**0**	725 617
Mushroom	1	**1.74**	24.06	**0**	32 828
	0.5	**3.62**	29.84	**0**	54 584
	0.1	**5.40**	41.38	**0**	100 528
	0.05	**6.37**	43.69	**0**	107 191
Pumsb	80	**133.97**	OOM	**0**	OOM
	75	**271**	OOM	**0**	OOM
	70	**509.79**	OOM	**0**	OOM

Backtrack-free

ClosedPattern: Experiments

ClosedPattern-DC vs ClosedPattern-WC :

Dataset	θ	Times(s)		#Failures	
	%	DC	WC	DC	WC
Chess	30	**45.92**	109.36	**0**	3×10^7
	20	**187.89**	480.52	**0**	1.3×10^8
	10	**969.40**	2288	**0**	6.3×10^8
Mushroom	1	**1.74**	3.42	**0**	1 117 883
	0.5	**3.62**	5.00	**0**	1 863 542
	0.1	**5.40**	9.96	**0**	3 401 297
	0.05	**6.37**	9.94	**0**	3 787 678
Pumsb	80	**133.97**	640.59	**0**	51 564
	75	**271**	1010.4	**0**	131 551
	70	**509.79**	2150.8	**0**	239 623

Third rule reduces considerably the explored nodes

ClosedPattern: Experiments

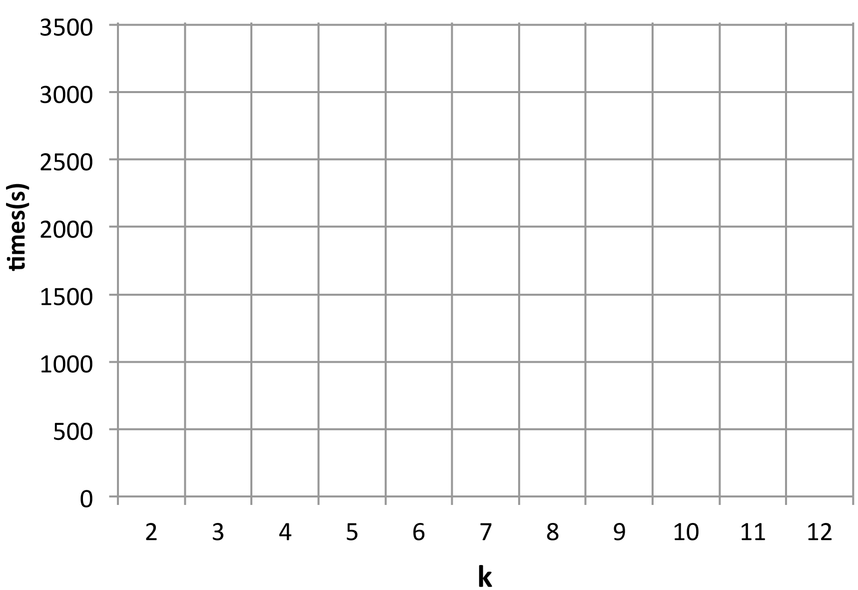
ClosedPattern-DC vs CP4IM vs LCM:

Dataset	θ	Times(s)		
	%	DC	CP4IM	LCM
Chess	30	45.92	136.31	**6.07**
	20	187.89	467.52	**27.55**
	10	969.40	1950.5	**141.55**
Mushroom	1	1.74	24.06	**0.22**
	0.5	3.62	29.84	**0.34**
	0.1	5.40	41.38	**0.47**
	0.05	6.37	43.69	**0.51**
Pumsb	80	133.97	OOM	**0.33**
	75	271	OOM	**0.48**
	70	509.79	OOM	**0.69**

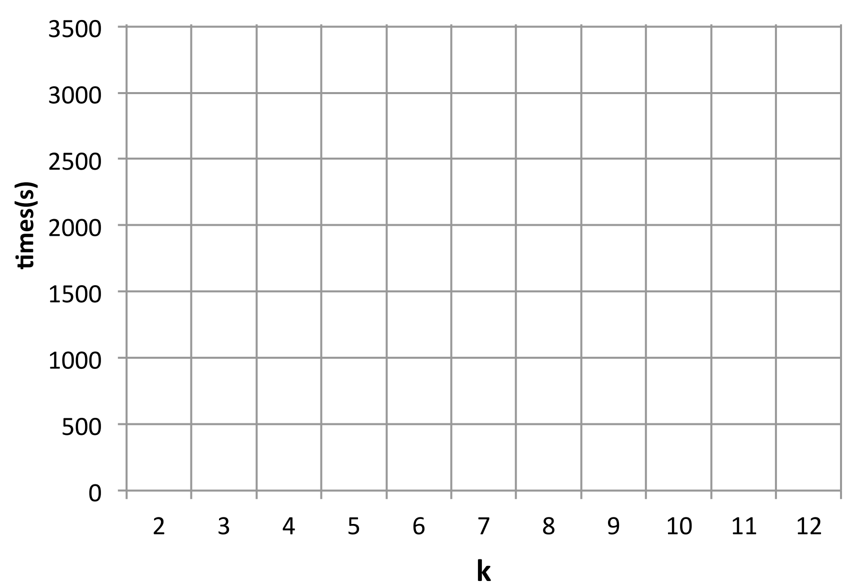
ClosedPattern: Experiments
k itemsets instance

The aim : find *k* closed itemsets:

(i) ClosedPattern-DC
(ii) Distinct itemsets
(iii) *lb* < *size* < *ub*



chess (θ = 80%, *lb* = 2, *ub* = 10)



connect (θ = 90%, *lb* = 2, *ub* = 10)

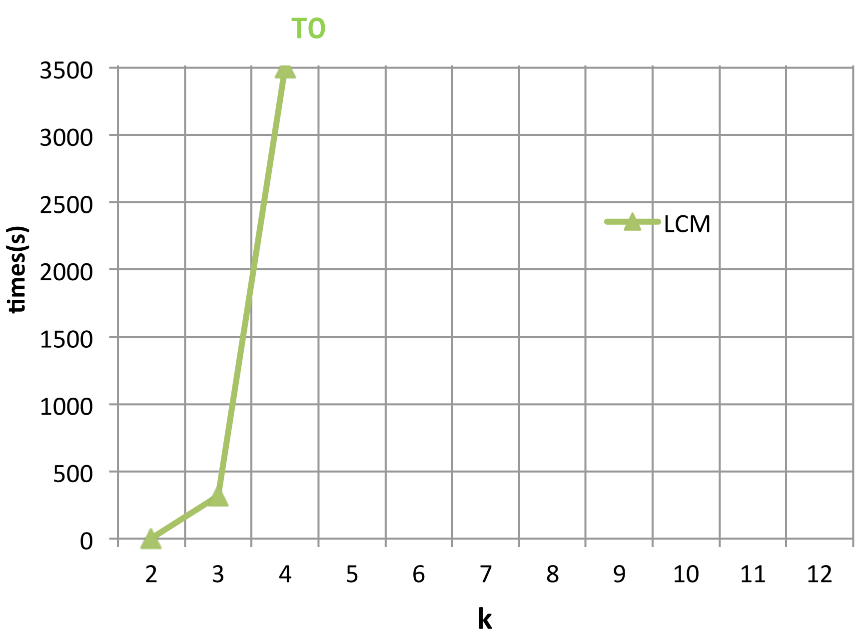
CLOSEDPATTERN: Experiments
k itemsets instance

The aim : find *k* closed itemsets:

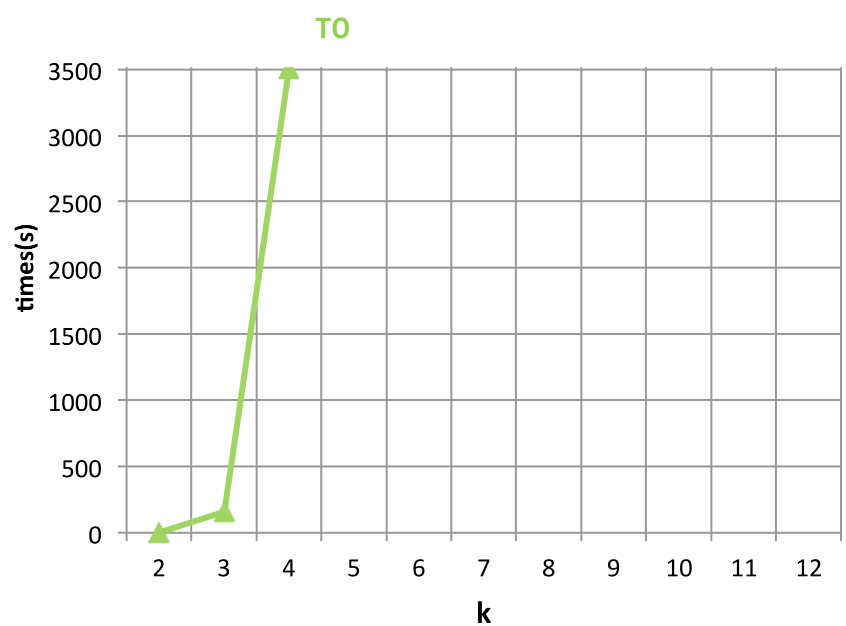
(i) CLOSEDPATTERN-DC

(ii) Distinct itemsets

(iii) *lb* < *size* < *ub*



chess (θ = 80%, *lb* = 2, *ub* = 10)



connect (θ = 90%, *lb* = 2, *ub* = 10)

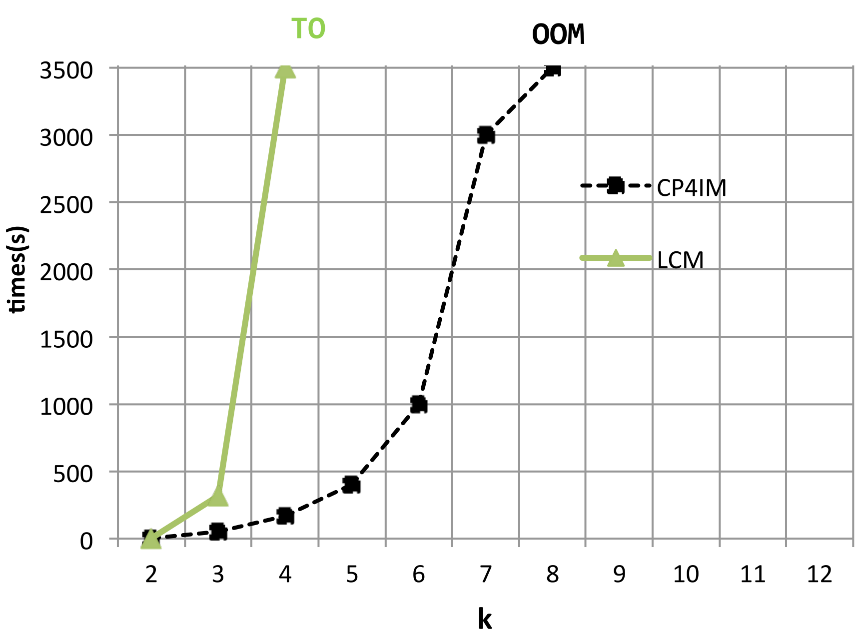
CLOSEDPATTERN: Experiments
k itemsets instance

The aim : find *k* closed itemsets:

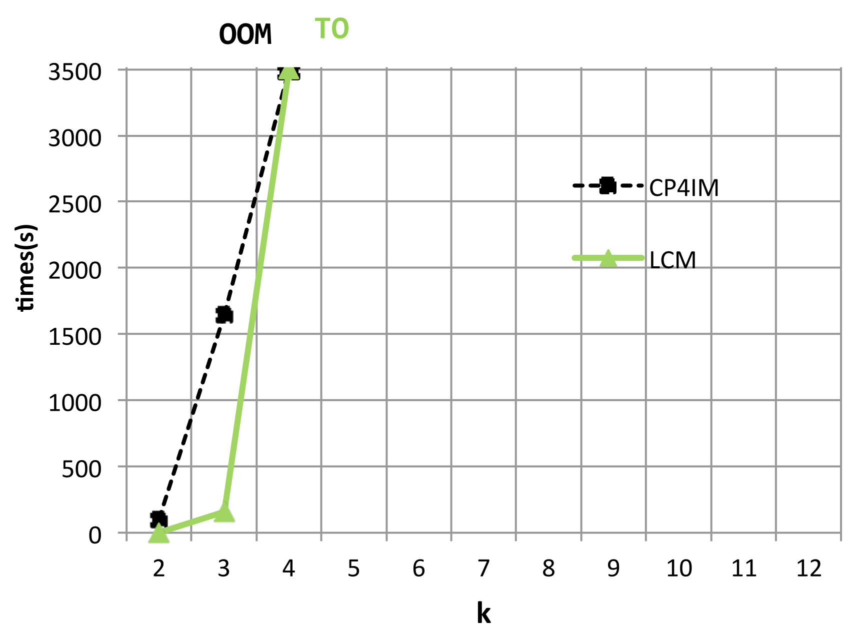
(i) CLOSEDPATTERN-DC

(ii) Distinct itemsets

(iii) *lb* < *size* < *ub*



chess (θ = 80%, *lb* = 2, *ub* = 10)



connect (θ = 90%, *lb* = 2, *ub* = 10)

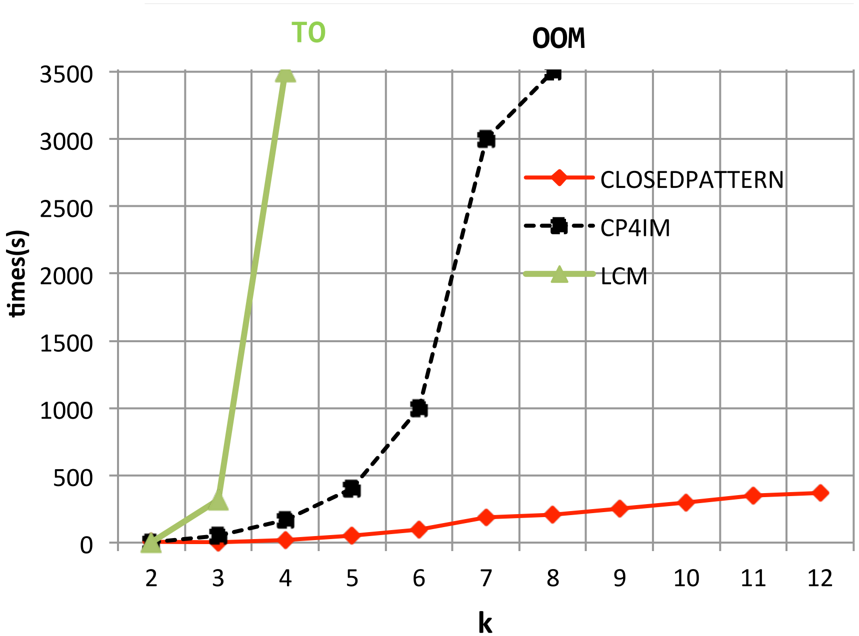
CLOSEDPATTERN: Experiments
k itemsets instance

The aim : find *k* closed itemsets:

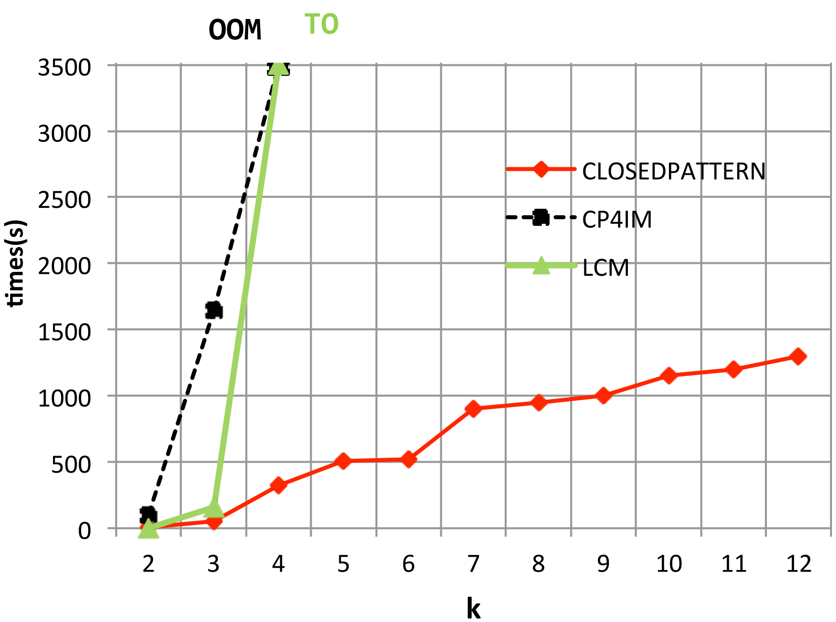
(i) CLOSEDPATTERN-DC

(ii) Distinct itemsets

(iii) *lb* < *size* < *ub*



chess (θ = 80%, *lb* = 2, *ub* = 10)



connect (θ = 90%, *lb* = 2, *ub* = 10)

Conclusion and perspectives

- Conclusion:

- A global constraint for Closed Frequent Itemset ensuring DC

- No need for reified constraints/extra variables

- Filtering algorithm cubic in time, quadratic in space.