K_SP: A resolution-based prover for multimodal K

Cláudia Nalon
nalon@unb.br
University of Brasília

Ullrich Hustadt
U.Hustadt@liverpool.ac.uk

Clare Dixon
C.Dixon@liverpool.ac.uk

University of Liverpool

IJCAR, 2016

Language

- ✔ K_n, the smallest multi-modal normal logic, extends propositional logic with a fixed, finite set of modal operators.
- ✔ The *set of well-formed formulae*, WFF_{K_n}, is the least set such that:
  - ✘ $p \in \mathcal{P} = \{p, q, p', q', p_1, q_1, \ldots\}$ and **true** are in WFF_{K_n};
  - ✘ if φ and ψ are in WFF_{K_n}, then so are $\neg\varphi$, $(\varphi \wedge \psi)$, and $\boxed{a}\varphi$ for each $a \in \mathcal{A}_n = \{1, \ldots, n\}$.
- ✔ Formulae are interpreted, as usual, with respect to Kripke structures:

$$\mathcal{M} = \langle \mathcal{W}, w_0, \mathcal{R}_1, \ldots, \mathcal{R}_n, \pi \rangle$$

where

$\langle \mathcal{M}, w \rangle \models \boxed{a}\varphi$ if, and only if, for all w', $w\mathcal{R}_a w'$ implies $\langle \mathcal{M}, w' \rangle \models \varphi$.

- ✔ Abbreviations: **false** $= \neg$**true**, $(\varphi \vee \psi) = \neg(\neg\varphi \wedge \neg\psi)$, $(\varphi \rightarrow \psi) = (\neg\varphi \vee \psi)$, and $\Diamond_a \varphi = \neg\boxed{a}\neg\varphi$.

Reasoning Tasks

$$\langle \mathcal{W}, w_0, \mathcal{R}_1, \ldots, \mathcal{R}_n, \pi \rangle$$

- ✔ A formula φ is locally satisfied in $\mathcal{M}$, denoted by $\mathcal{M} \models_L \varphi$, if $\langle \mathcal{M}, w_0 \rangle \models \varphi$.
- ✔ A formula φ is locally satisfiable if there is a model $\mathcal{M}$ such that $\mathcal{M}$ locally satisfies φ.
- ✔ A formula φ is globally satisfied in $\mathcal{M}$, denoted by $\mathcal{M} \models_G \varphi$, if for all $w \in \mathcal{W}$, $\langle \mathcal{M}, w \rangle \models \varphi$.
- ✔ A formula φ is globally satisfiable if there is a model $\mathcal{M}$ such that $\mathcal{M}$ globally satisfies φ.
- ✔ Given a set of formulae Γ and a formula φ, the local satisfiability of φ under the global constraints Γ consists of showing that there is a model that globally satisfies the formulae in Γ and that there is a world in this model that satisfies φ.

Complexity & Proof Methods

- ✔ Complexity:
  - ✘ Local satisfiability: PSPACE-complete;
  - ✘ Global satisfiability: EXPTIME-complete;
  - ✘ Local satisfiability under global constraints: EXPTIME-complete.
- ✔ Proof Methods:
  - ✘ Translation into first-order logic;
  - ✘ Sequent calculus;
  - ✘ Tableaux;
  - ✘ Inverse method;
  - ✘ BDD;
  - ✘ SAT;
  - ✘ Resolution;
  - ✘ ...

Example

$$\Diamond \Diamond p \wedge \Box \neg p$$

Example

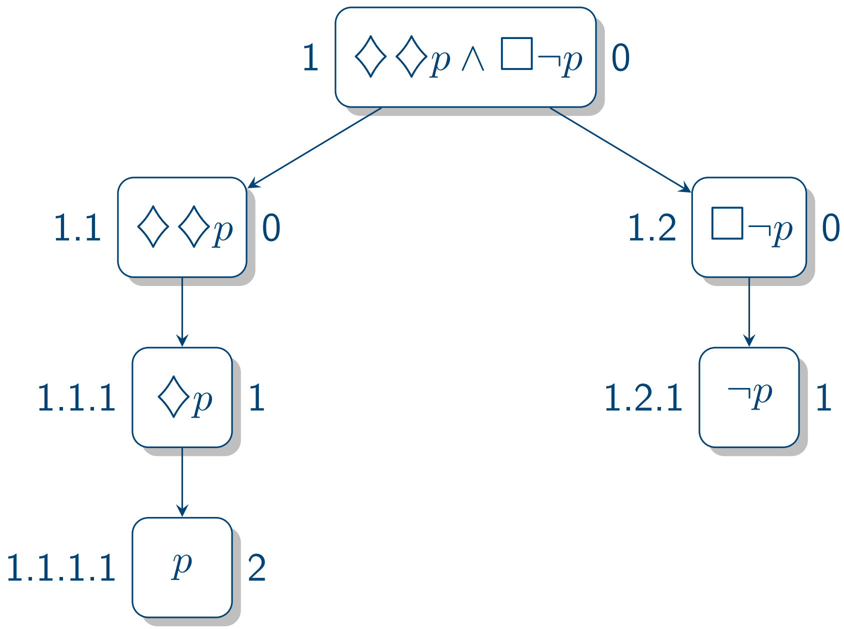
$\Diamond\Diamond p \wedge \Box\neg p$

1. $\textbf{start} \rightarrow t_0$
2. $\boxed{*}(t_0 \rightarrow \Diamond t_1)$
3. $\boxed{*}(t_1 \rightarrow \Diamond p)$
4. $\boxed{*}(t_0 \rightarrow \Box\neg p)$

Example

$$\Diamond\Diamond p \wedge \Box\neg p$$

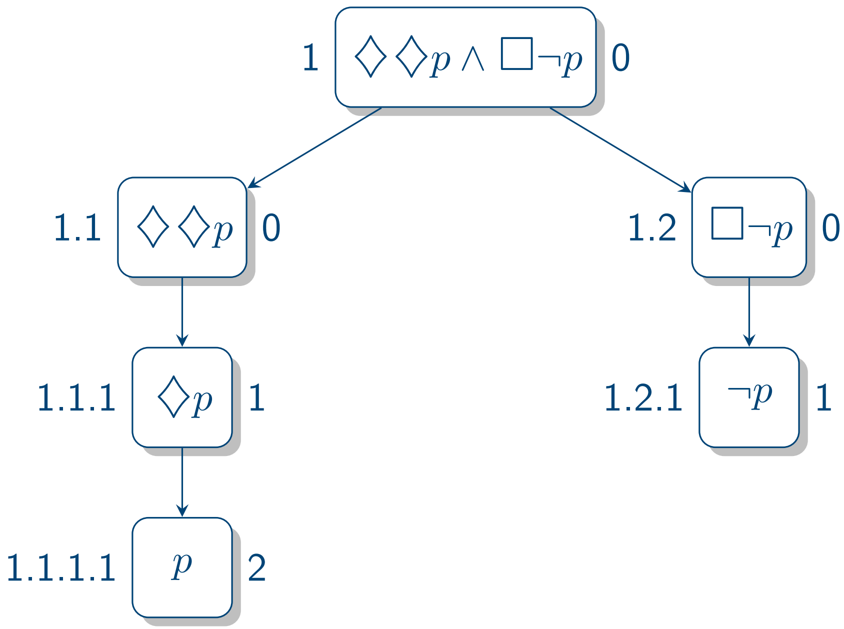
1. $\mathbf{start} \rightarrow t_0$
2. $\boxed{*}(t_0 \rightarrow \Diamond t_1)$
3. $\boxed{*}(t_1 \rightarrow \Diamond p)$
4. $\boxed{*}(t_0 \rightarrow \Box\neg p)$

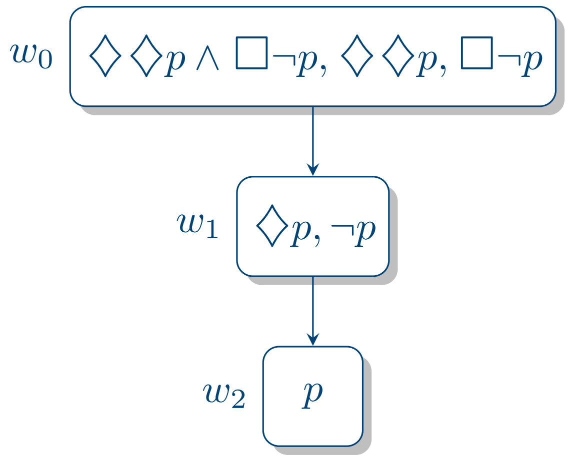


Example

$$\Diamond\Diamond p \wedge \Box\neg p$$

1. **start** $\rightarrow t_0$
2. $\boxed{*}(t_0 \rightarrow \Diamond t_1)$
3. $\boxed{*}(t_1 \rightarrow \Diamond p)$
4. $\boxed{*}(t_0 \rightarrow \Box\neg p)$





Previous work - I

✔ Areces, C., Gennari, R., Heguiabehere, J., de Rijke, M.: Tree-based heuristics in modal theorem proving. In: Proc. of ECAI 2000. pp. 199-203. IOS Press (2000).

$$\Diamond\Diamond p \wedge \Box\neg p \Longrightarrow \Diamond\Diamond p_2 \wedge \Box\neg p_1$$

Previous work - I

- ✔ Areces, C., Gennari, R., Heguiabehere, J., de Rijke, M.: Tree-based heuristics in modal theorem proving. In: Proc. of ECAI 2000. pp. 199-203. IOS Press (2000).

$$\Diamond\Diamond p \wedge \Box\neg p \Longrightarrow \Diamond\Diamond p_2 \wedge \Box\neg p_1$$

$$p \wedge \Diamond\neg p \Longrightarrow p_0 \wedge \Diamond\neg p_1$$

Previous work - II

✔ Areces, C., de Nivelle, H., de Rijke, M.: Prefixed Resolution: A Resolution Method for Modal and Description Logics. In: Ganzinger, H. (ed.) Proc. CADE-16. LNAI, vol. 1632, pp. 187-201. Springer, Berlin (Jul 7-10 1999).

- ✘ Formulae labelled by either constants or pair of constants.
- ✘ The inference rule for $\diamond$ generates new labels.
- ✘ The inference rule for $\square$ corresponds to propagation.

For instance: $Tr(\diamond p \wedge \diamond q \wedge \square((p \vee q) \wedge (\neg p \vee \neg q) \wedge (\neg p \vee q)), i)$ would lead to:

$$\begin{array}{ll|ll}
1.i, 1: & p & 1'.i, 2: & q \\
2.i, 1: & p \vee q & 2'.i, 2: & p \vee q \\
3.i, 1: & \neg p \vee \neg q & 3'.i, 2: & \neg p \vee \neg q \\
4.i, 1: & \neg p \vee q & 4'.i, 2: & \neg p \vee q
\end{array}$$

which might result in applying the same resolution steps to clauses in $\{2, 3, 4\}$ and $\{2', 3', 4'\}$.

The main idea

- ✔ The calculus should allow for both local and modal reasoning.
- ✔ A formula to be tested for (un)satisfiability is translated into a normal form, where labels refer to the modal level they occur.
- ✔ Inference rules are then applied by modal level.

Clauses

- ✔ Literal clause $ml : \bigvee_{b=1}^{r} l_b$
- ✔ Positive a-clause $ml : l' \to \boxed{a} l$
- ✔ Negative a-clause $ml : l' \to \Diamond_a l$

where $ml \in \mathbb{N} \cup \{*\}$ and l, l', $l_b \in \mathcal{L}$. Positive and negative a-clauses are together known as *modal a-clauses*; the index a may be omitted if it is clear from the context.

Examples

$$\Diamond \Diamond p \wedge \Box \neg p$$

$$0: \quad t_0$$
$$0: \quad t_0 \rightarrow \Diamond t_1$$
$$1: \quad t_1 \rightarrow \Diamond p$$
$$0: \quad t_0 \rightarrow \Box \neg p$$

Examples

$$\Diamond\Diamond p \wedge \Box\neg p$$

$$\begin{array}{ll} 0: & t_0 \\ 0: & t_0 \rightarrow \Diamond t_1 \\ 1: & t_1 \rightarrow \Diamond p \\ 0: & t_0 \rightarrow \Box\neg p \end{array}$$

$$p \wedge \Diamond\neg p$$

$$\begin{array}{ll} 0: & t_0 \\ 0: & \neg t_0 \vee p \\ 0: & t_0 \rightarrow \Diamond\neg p \end{array}$$

Examples

$$\Diamond\Diamond p \wedge \Box\neg p$$

$$\begin{array}{ll} 0: & t_0 \\ 0: & t_0 \rightarrow \Diamond t_1 \\ 1: & t_1 \rightarrow \Diamond p \\ 0: & t_0 \rightarrow \Box\neg p \end{array}$$

$$p \wedge \Diamond\neg p$$

$$\begin{array}{ll} 0: & t_0 \\ 0: & \neg t_0 \vee p \\ 0: & t_0 \rightarrow \Diamond\neg p \end{array}$$

$$p \wedge \Diamond\neg p$$

$$\begin{array}{ll} *: & t_0 \\ *: & \neg t_0 \vee p \\ *: & t_0 \rightarrow \Diamond\neg p \end{array}$$

Inference Rules

[LRES]

$$\begin{array}{rccc} ml: & D & \vee & l \\ ml': & D' & \vee & \neg l \\ \hline \sigma(\{ml, ml'\}): & D & \vee & D' \end{array}$$

[MRES]

$$\begin{array}{rccc} ml: & l_1 & \rightarrow & \boxed{a} l \\ ml': & l_2 & \rightarrow & \diamond_a \neg l \\ \hline \sigma(\{ml, ml'\}): & \neg l_1 & \vee & \neg l_2 \end{array}$$

Inference Rules

[GEN1]

$$
\begin{array}{rlll}
ml_1: & l'_1 & \rightarrow & \boxed{a}\neg l_1 \\
 & \vdots & & \\
ml_m: & l'_m & \rightarrow & \boxed{a}\neg l_m \\
ml_{m+1}: & l' & \rightarrow & \Diamond_a \neg l \\
ml_{m+2}: & \multicolumn{3}{l}{l_1 \vee \ldots \vee l_m \vee l} \\
\hline
ml: & \multicolumn{3}{l}{\neg l'_1 \vee \ldots \vee \neg l'_m \vee \neg l'}
\end{array}
$$

where $ml = \sigma(\{ml_1, \ldots, ml_{m+1}, ml_{m+2} - 1\})$

Inference Rules

[GEN1]

$$\begin{array}{rcl} ml_1 : & l'_1 & \rightarrow \boxed{a}\neg l_1 \\ & \vdots & \\ ml_m : & l'_m & \rightarrow \boxed{a}\neg l_m \\ ml_{m+1} : & l' & \rightarrow \Diamond_a \neg l \\ ml_{m+2} : & l_1 \vee \ldots \vee l_m \vee l & \\ \hline ml : & \neg l'_1 \vee \ldots \vee \neg l'_m \vee \neg l' & \end{array}$$

where $ml = \sigma(\{ml_1, \ldots, ml_{m+1}, ml_{m+2} - 1\})$

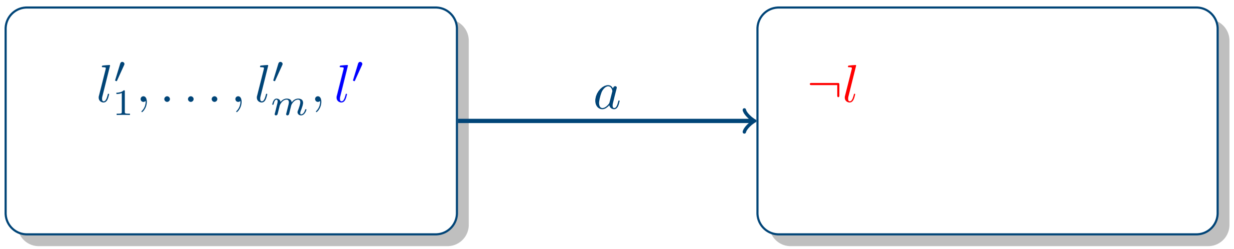
$l'_1, \ldots, l'_m, l'$

Inference Rules

[GEN1]

$$
\begin{array}{rl}
ml_1 : & l'_1 \rightarrow [a]\neg l_1 \\
& \vdots \\
ml_m : & l'_m \rightarrow [a]\neg l_m \\
ml_{m+1} : & l' \rightarrow \langle a \rangle \neg l \\
ml_{m+2} : & l_1 \vee \ldots \vee l_m \vee l \\
\hline
ml : & \neg l'_1 \vee \ldots \vee \neg l'_m \vee \neg l'
\end{array}
$$

where $ml = \sigma(\{ml_1, \ldots, ml_{m+1}, ml_{m+2} - 1\})$

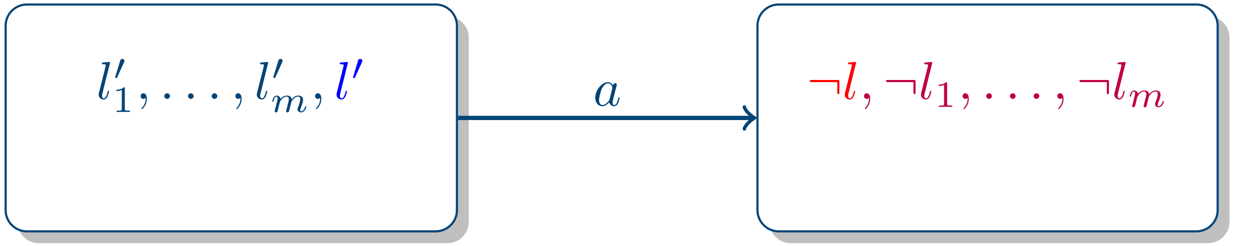


Inference Rules

[GEN1]

$$\begin{array}{rl} ml_1: & l'_1 \rightarrow \boxed{a}\neg l_1 \\ & \vdots \\ ml_m: & l'_m \rightarrow \boxed{a}\neg l_m \\ ml_{m+1}: & l' \rightarrow \Diamond_a \neg l \\ ml_{m+2}: & l_1 \vee \ldots \vee l_m \vee l \\ \hline ml: & \neg l'_1 \vee \ldots \vee \neg l'_m \vee \neg l' \end{array}$$

where $ml = \sigma(\{ml_1, \ldots, ml_{m+1}, ml_{m+2} - 1\})$

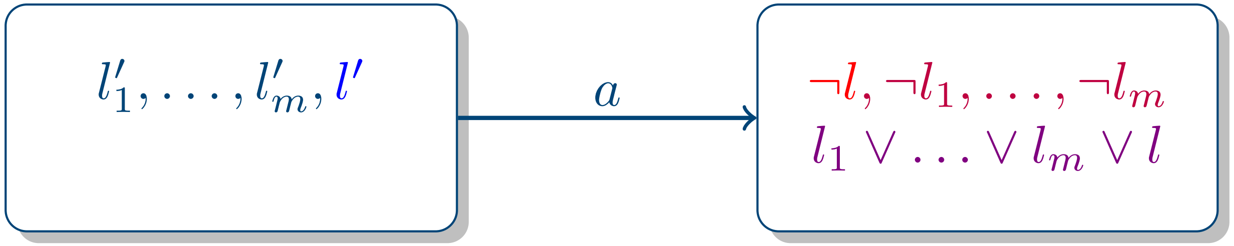


Inference Rules

[GEN1]

$$
\begin{array}{rl}
ml_1 : & l'_1 \rightarrow \boxed{a}\neg l_1 \\
& \vdots \\
ml_m : & l'_m \rightarrow \boxed{a}\neg l_m \\
ml_{m+1} : & l' \rightarrow \Diamond_a \neg l \\
ml_{m+2} : & l_1 \vee \ldots \vee l_m \vee l \\
\hline
ml : & \neg l'_1 \vee \ldots \vee \neg l'_m \vee \neg l'
\end{array}
$$

where $ml = \sigma(\{ml_1, \ldots, ml_{m+1}, ml_{m+2} - 1\})$



Inference Rules

[GEN2]

$$\begin{array}{rlcl} ml_1 : & l'_1 & \rightarrow & \boxed{a} l_1 \\ ml_2 : & l'_2 & \rightarrow & \boxed{a} \neg l_1 \\ ml_3 : & l'_3 & \rightarrow & \Diamond_a l_2 \\ \hline ml : & \multicolumn{3}{c}{\neg l'_1 \vee \neg l'_2 \vee \neg l'_3} \end{array}$$

where $ml = \sigma(\{ml_1, ml_2, ml_3\})$

[GEN3]

$$\begin{array}{rlcl} ml_1 : & l'_1 & \rightarrow & \boxed{a} \neg l_1 \\ & \vdots & & \\ ml_m : & l'_m & \rightarrow & \boxed{a} \neg l_m \\ ml_{m+1} : & l' & \rightarrow & \Diamond_a l \\ ml_{m+2} : & \multicolumn{3}{l}{l_1 \vee \ldots \vee l_m} \\ \hline ml : & \multicolumn{3}{l}{\neg l'_1 \vee \ldots \vee \neg l'_m \vee \neg l'} \end{array}$$

where $ml = \sigma(\{ml_1, \ldots, ml_{m+1}, ml_{m+2} - 1\})$

Negative Resolution

- ✔ For completeness, we need to change the normal form:

$$\begin{aligned}\rho(ml : t \to \boxed{a}\neg p) &= (ml : t \to \boxed{a} t') \wedge \rho(ml+1 : t' \to \neg p)\\ \rho(ml : t \to \Diamond_a \neg p) &= (ml : t \to \Diamond_a t') \wedge \rho(ml+1 : t' \to \neg p)\end{aligned}$$

- ✔ The inference rule is now of the form:

[NEG LRES]

$$\frac{\begin{array}{rlll} ml : & D & \vee & l \\ ml' : & D' & \vee & \neg l \end{array}}{\begin{array}{rlll}\sigma(\{ml, ml'\}) : & D & \vee & D'\end{array}}$$

where one of the premises is negative.

Ordered Resolution

- ✔ Assumes a total ordering on the literals occurring in the normal form.
- ✔ Normal form is also extended in order to achieve completeness: generating *small literals*.
- ✔ The inference rule takes the form:

$$\begin{array}{rccc} \multicolumn{4}{c}{\text{[ORD LRES]}} \\ ml: & D & \vee & l \\ ml': & D' & \vee & \neg l \\ \hline \sigma(\{ml, ml'\}): & D & \vee & D' \end{array}$$

where l is strictly maximal with respect to D and $\neg l$ is strictly maximal with respect to D'.

Implementation

- ✔ Set-of-support (given-clause, as in Otter), but there is one set of support for each modal level;
- ✔ Refinements: negative, ordered, negative+ordered, ordered with selection, positive resolution;
- ✔ Pre-processing: simplification, pure literal elimination, modal level pure literal elimination, unit propagation, populating automatically the usable, different techniques for renaming, prenex/antiprenex, cnf, small cnf (in progress);
- ✔ Redundancy elimination: (lazy) forward/backward subsumption, pure literal elimination, modal level pure literal elimination . . .
- ✔ Clause selection: shortest, newest, oldest, greatest literal, smallest literal;
- ✔ The full pack is in my webpage.

KSP - LWB - k_t4p - Modal Layering

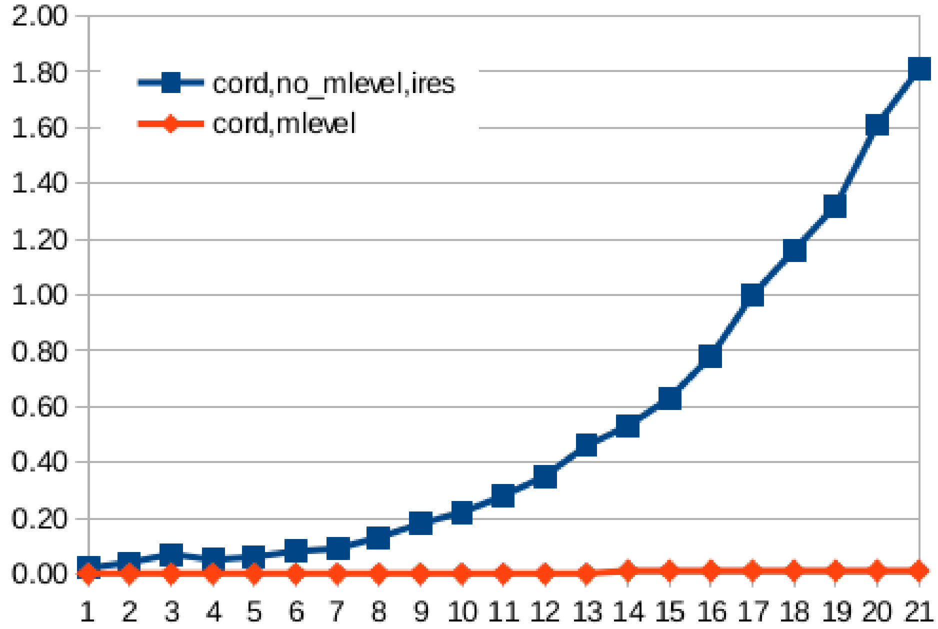


Figure 1: Unsatisfiable Formulae

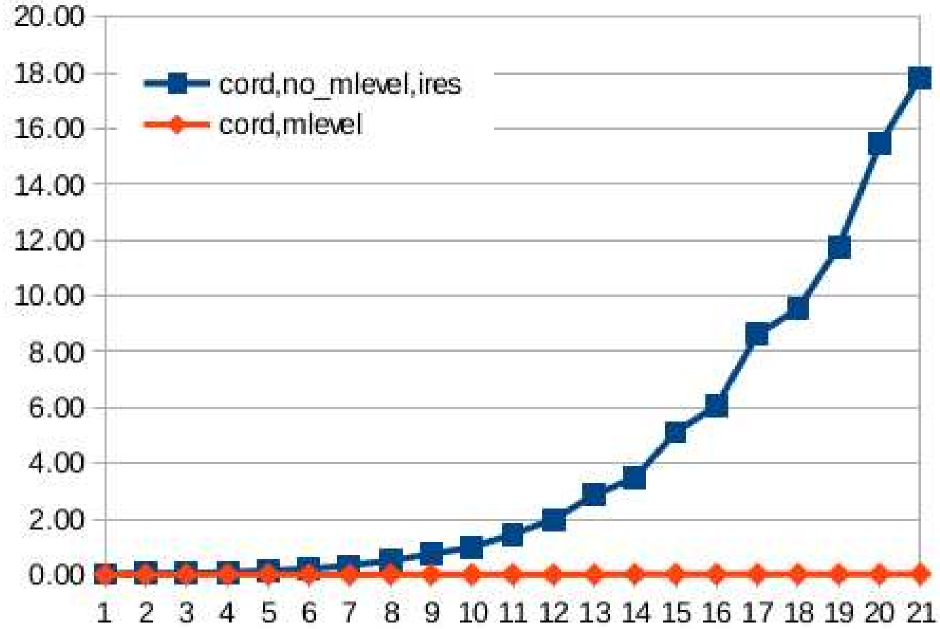
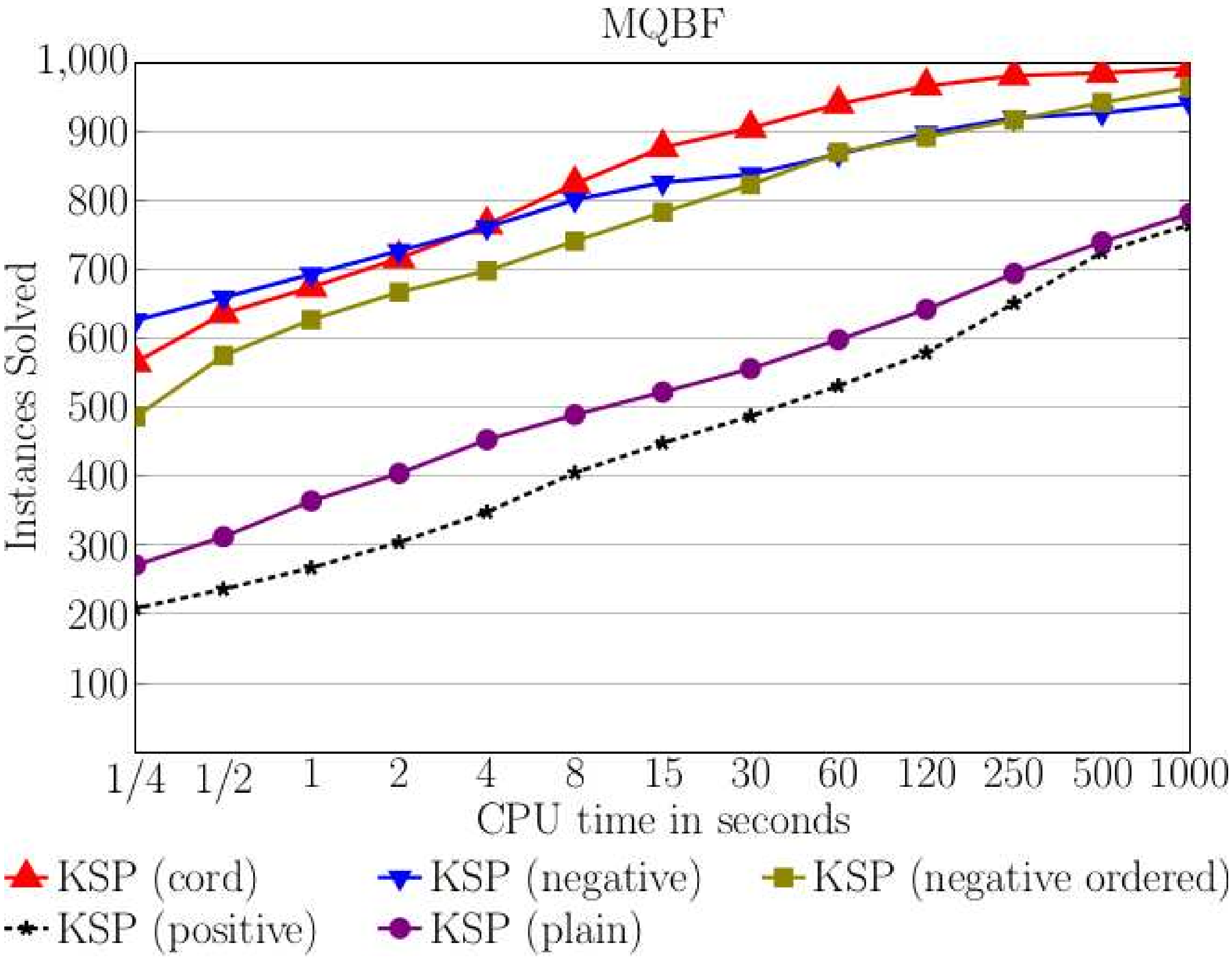
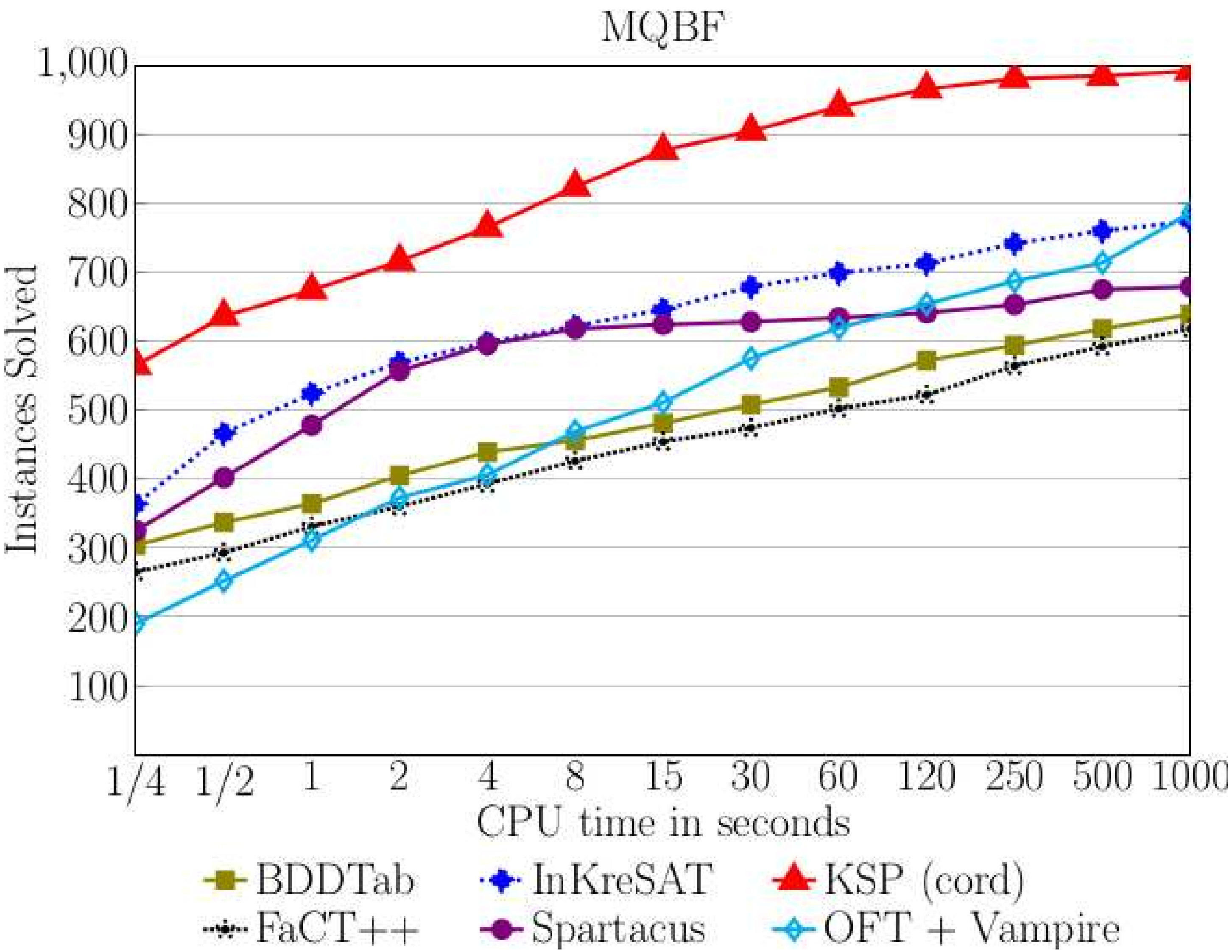


Figure 2: Satisfiable Formulae

KSP - MQBF - Different Refinements

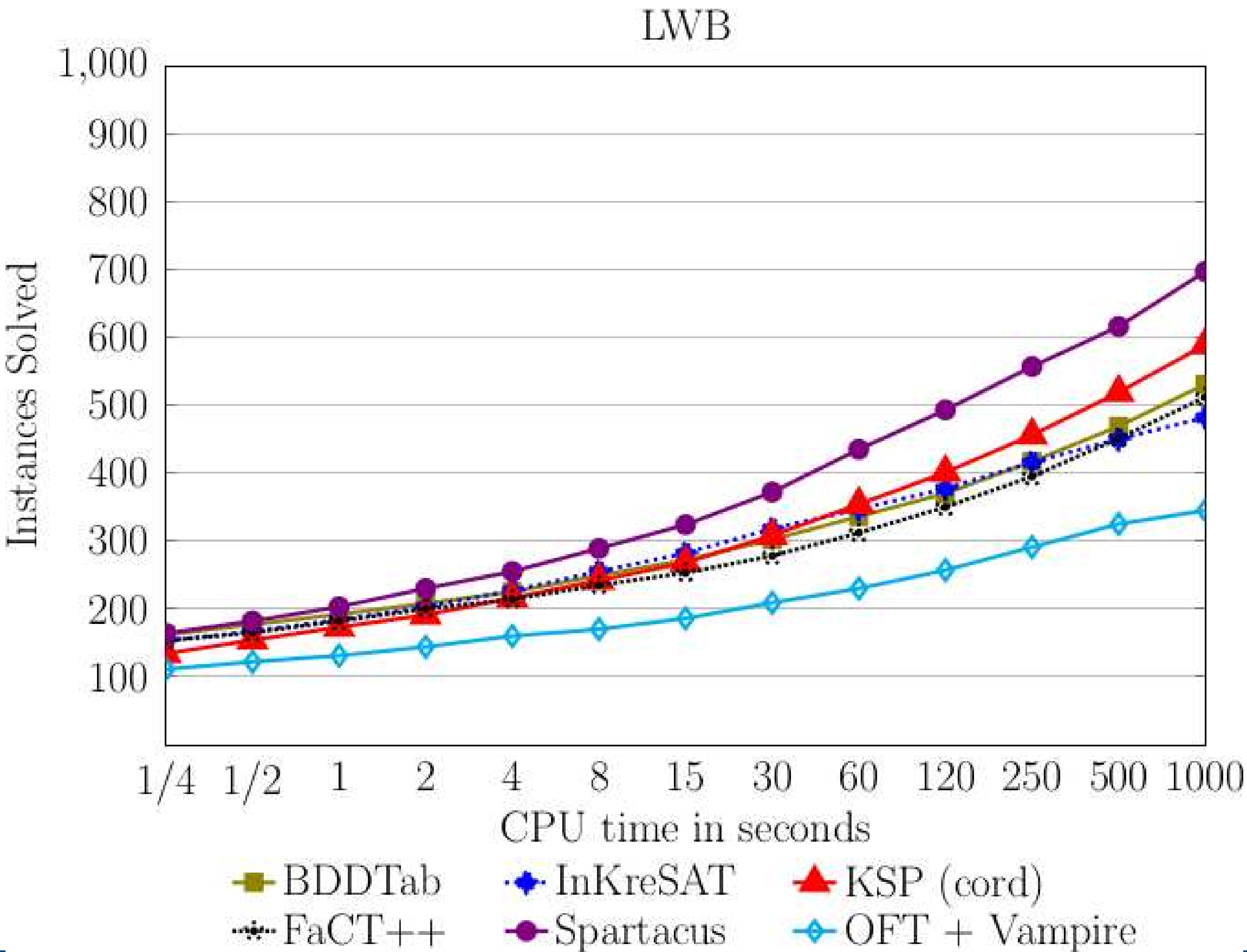



MQBF
Instances Solved
1,000
900
800
700
600
500
400
300
200
100
1/4 1/2 1 2 4 8 15 30 60 120 250 500 1000
CPU time in seconds
BDDTab
InKreSAT
KSP (cord)
FaCT++
Spartacus
OFT + Vampire

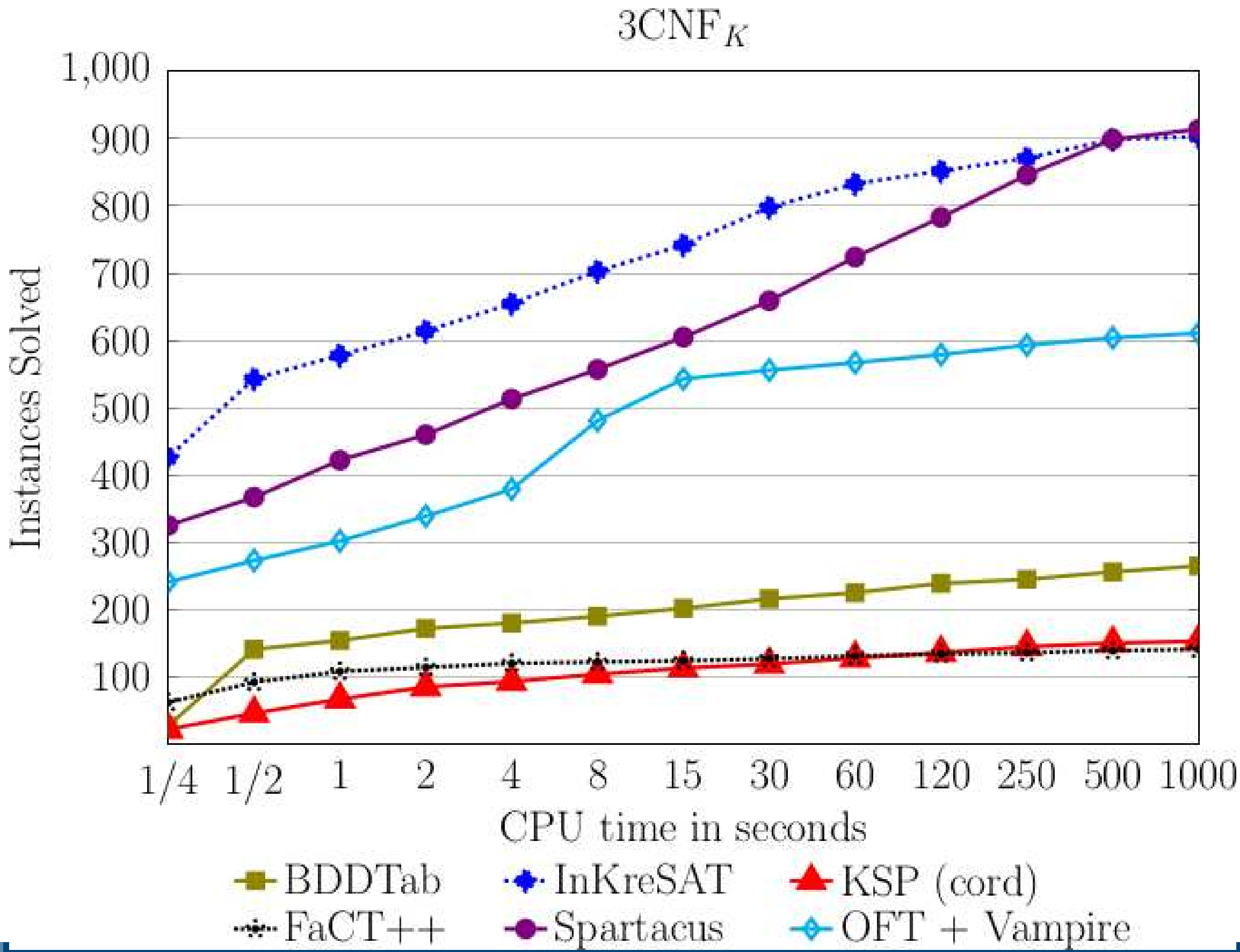
	BDDTab		FaCT++		InKreSAT		K_SP (cord)		Spartacus		OFT + Vampire	
k_branch_n	22	*22*	12	*12*	15	*15*	18	*18*	12	*12*	**50**	*70*
k_branch_p	22	*22*	12	*12*	22	*22*	24	*24*	14	*14*	**50**	*70*
k_d4_n	20	*440*	6	*40*	34		**47**	*1520*	28	*760*	14	*200*
k_d4_p	26	*640*	24	*600*	18	*360*	**54**	*1800*	32	*920*	21	*960*
k_dum_n	39	*2400*	42	*2640*	23	*1120*	**51**	*3360*	44	*2800*	17	*640*
k_dum_p	42	*2640*	38	*2320*	28	*1520*	**51**	*3360*	46	*2960*	18	*720*
k_grz_n	35	*2600*	27	*1800*	50	*4500*	5	*50*	**52**	*5500*	24	*1500*
k_grz_p	35	*2600*	27	*1800*	51	*5000*	51	*5000*	**52**	*5500*	27	*1800*
k_lin_n	46	*4000*	43	*3400*	33	*2500*	1	*10*	**50**	*4800*	40	*3100*
k_lin_p	14	*500*	28	*10000*	**56**	*500000*	23	*5000*	55	*400000*	28	*10000*
k_path_n	37	*290*	48	*400*	7	*14*	**53**	*900*	48	*400*	41	*330*
k_path_p	35	*270*	48	*400*	5	*12*	**53**	*900*	48	*400*	41	*330*
k_ph_n	10	*10*	8	*16*	24	*90*	3	*6*	**21**	*75*	15	*45*
k_ph_p	**11**	*11*	9	*8*	5	*5*	5	*5*	9	*9*	10	*10*
k_poly_n	39	*600*	34	*500*	30		34	*500*	**44**	*720*	20	*220*
k_poly_p	38	*580*	34	*500*	28	*400*	34	*500*	**44**	*700*	20	*220*
k_t4p_n	40	*3500*	24	*1500*	17	*800*	36	*2700*	**45**	*6000*	11	*200*
k_t4p_p	48	*7500*	49	*8000*	28		46	*6500*	**53**	*12000*	14	*500*

Table 1: Detailed benchmarking results on LWB

All Provers - LWB



All Provers - 3CNF



Conclusion and Future Work

- ✔ The calculus for K_n was implemented and tested.
- ✔ Results are promising.
- ✔ Negative and ordered resolution, together with layering, are also complete.
- ✔ KSP is not any clever yet.

TABLEAUX 2017 FroCoS 2017 ITP 2017

Brasília, Brazil

25 – 29 September