Computational Complexity of Topological Logics

Ian Pratt-Hartmann

School of Computer Science

Manchester University

(Joint work with Roman Kontchakov and Michael Zakharyaschev.)

Summer Conference on Topology and its Applications

Brno, 14th–17th July, 2009

- A well-known fact:

  Suppose r_1 and r_2 are subsets of a topological space with r_1 included in r_2 and r_2 included in the closure of r_1. If r_1 connected, then so is r_2.

- We might write this in symbols as follows

$$c(r_1) \wedge r_1 \subseteq r_2 \wedge r_2 \subseteq r_1^- \rightarrow c(r_2).$$

- Once we are working in a formal language, the following issue become salient:
  - decidability and complexity
  - expresive power
  - axiomatizability
  - model theory.

- A topological language is a formal language with (object) variables $\mathbf{R} = \{r_1, r_2, \ldots\}$, whose non-logical signature is given a fixed 'topological' interpretation (e.g. $\mathbf{0}$, $^-$, c, $\subseteq$, $\ldots$).

- A topological frame is a pair $\mathfrak{F} = (T, W)$, where T is a topological space and $W \subseteq 2^T$. A topological interpretation is a pair $\mathfrak{A} = (\mathfrak{F}, \cdot^{\mathfrak{A}})$, where $\mathfrak{F} = (T, W)$ is a topological frame and $\cdot^{\mathfrak{A}}$ a function $\mathbf{R} \rightarrow W$.

- The notions of truth in an interpretation and satisfiability (dually: validity) over a frame (or class of frames) are understood as expected $\ldots$

- If $\mathcal{K}$ is a class of frames, we denote the satisfiability problem for $\mathcal{L}$-formulas over a class of frames $\mathcal{K}$ by $\mathcal{S}at(\mathcal{L}, \mathcal{K})$.

- The most basic class of frames to consider is:

$$\textsc{All} = \{(T, 2^T) \mid T \text{ a topological space}\}.$$

Are there any others we should look at?

- Recall that a regular closed set is one that is equal to the closure of its interior: $u = u^{\circ -}$.

- The regular closed sets of any topological space T form a Boolean algebra:

$$u + v = u \cup v, \qquad u \cdot v = (u \cap v)^{\circ -}, \qquad -u = (T \setminus u)^{-}$$

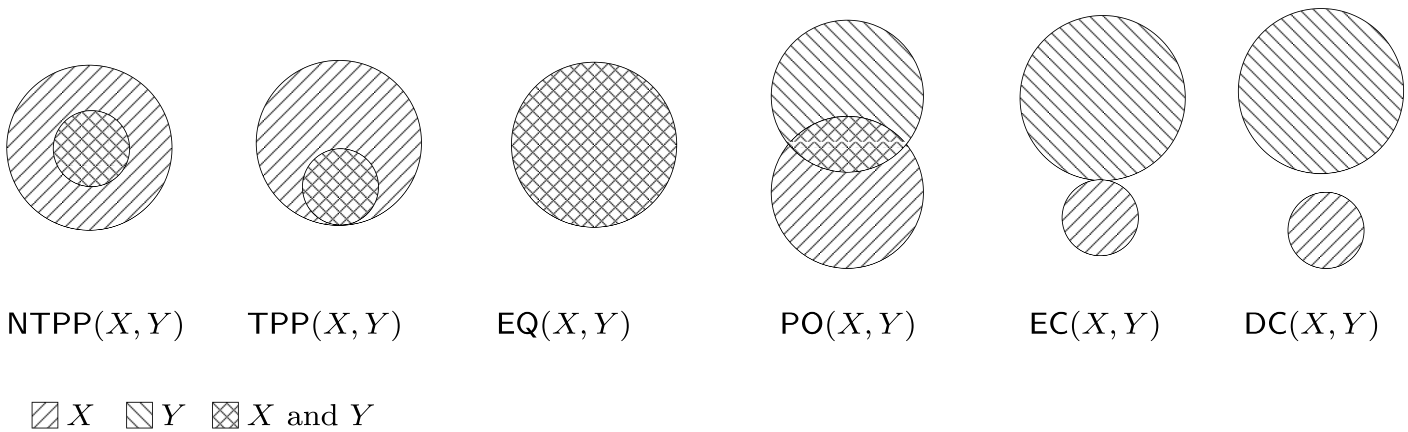
We denote the set of regular closed subsets of T by $\boldsymbol{RC}(T)$.

- We shall be interested in the frame class

$$\textsc{RegC} = \{(T, \boldsymbol{RC}(T)) \mid T \text{ a topological space}\}.$$

and some of its sub-classes.

- Perhaps the most intensively studied topological language is $\mathcal{RCC}$-8 (Randall, Cui, Cohn, 1992; Egenhoffer 1991):



- Example of a formula valid over RegC:

$$\big(\mathsf{TPP}(r_1, r_2) \wedge \mathsf{NTPP}(r_1, r_3)\big) \rightarrow \big(\mathsf{PO}(r_2, r_3) \vee \mathsf{TPP}(r_2, r_3) \vee \mathsf{NTPP}(r_2, r_3)\big)$$

- The problem $\mathcal{S}at(\mathcal{RCC}\text{-}8, \text{RegC})$ is NPTime-complete (Renz, 1998); the satisfiability problem for conjunctions of $\mathcal{RCC}$-8-literals is NLogSpace-complete (Griffiths, 2007).

- The language $\mathcal{C}$ (Wolter and Zakharyaschev, 2000) adds Boolean operators to this language, i.e. we have the primitives $\mathbf{0}$, $\mathbf{1}$, $+$, $\cdot$, $-$ (with the obvious interpretations), in addition to the $\mathcal{RCC}$-8-predicates.

- The following $\mathcal{C}$-formula is valid over RegC:

$$\mathsf{EC}(r_1 + r_2, r_3) \leftrightarrow \big(\mathsf{EC}(r_1, r_3) \vee \mathsf{EC}(r_2, r_3)\big)$$

- The problem $\mathcal{S}at(\mathcal{C}, \textsc{RegC})$ is still NPTime-complete.

- Using the function symbols $+$, $\cdot$ and $-$, we can replace the $\mathcal{RCC}$-8-predicates with the single binary relation of contact:

$$C(r_1, r_2) \text{ iff } r_1 \cap r_2 = \emptyset.$$

For instance, the literal $\mathsf{EC}(\tau_1, \tau_2)$ can be expressed as

$$C(\tau_1, \tau_2) \wedge \tau_1 \cdot \tau_2 = \mathbf{0}.$$

- A more familiar example: let $\mathcal{S}4_u$ be the topological language with primitives
  - $\mathbf{0}$, $\mathbf{1}$
  - $^-$, $^\circ$ (topological interior)
  - $\cup$, , $-$ (complement in whole space).
- The following $\mathcal{S}4_u$-formula is valid over the class All.

$$r_1 \cap r_2^\circ = \mathbf{0} \rightarrow r_1^- \cap r_2^\circ = \mathbf{0}$$

- We may write it in the syntax of modal logic, under the topological semantics of McKinsey and Tarski (1944):

$$\mathcal{U}\neg(p_1 \wedge \Box p_2) \rightarrow \mathcal{U}\neg(\Diamond p_1 \wedge \Box p_2).$$

- The satisfiability problem for this logic is known to be PSpace-complete (Ladner, 1977).

- None of the languages in the previous summary features the convexity predicate c.
- Define the languages $\mathcal{RCC}$-8c, $\mathcal{C}c$ and $\mathcal{S}4_u c$ by adding c to $\mathcal{RCC}$-8, $\mathcal{C}$ and $\mathcal{S}4_u$, repectively.
  - The following $\mathcal{C}c$-formula is valid over RegC:

$$c(r_1) \wedge c(r_2) \wedge C(r_1, r_2) \rightarrow c(r_1 + r_2).$$

  - The following $\mathcal{S}4_u c$-formula is valid over All:

$$c(r_1) \wedge r_1 \subseteq r_2 \wedge r_2 \subseteq r_1^- \rightarrow c(r_2).$$

- More ambitiously, we can add the predicates
  - $c^{\leq k}$ (has at most k components)
  - $c^{\geq k}$ (has at least k components)

  for all $k \geq 0$.
- This gives us the languages $\mathcal{RCC}$-8cc, $\mathcal{C}cc$ and $\mathcal{S}4_u cc$.
  - The following $\mathcal{C}cc$-formula is valid over RegC:

$$\left(c^{\leq k}(r_1) \ \wedge \ c^{\leq \ell}(r_2) \ \wedge \ C(r_1, r_2)\right) \ \rightarrow \ c^{\leq \ell + k - 1}(r_1 + r_2).$$

- What can we say about the complexity of the satisfiability problems for these langauges?

- The language $\mathcal{C}c$ exhibits some interesting behaviour: let $\mathcal{C}c^1$ be the restriction of $\mathcal{C}c$ to formulas containing at most one one occurrence of c.

- Given the variables $r_n, \ldots, r_1$, we can number the various terms $\pm r_n \cdot \cdots \cdot \pm r_1$ as

$$\tau_0 \cdot \cdots \quad \cdots \quad \cdots \cdot \tau_{2^n-1}.$$

We may assume that, in the term τ_j $(0 \leq j < 2^n)$, the ith factor (i.e. r_1 or $-r_i$) specifies the ith bit of j.

- We can write a $\mathcal{C}c$-formula enforcing the condition

$$\neg C(\tau_i, \tau_j) \qquad \text{for all } 0 \leq i < j < 2^n \text{ with } j - i > 1.$$

with the size of this formula polynomial in n.

- Thus, the $\mathcal{C}c^1$-formula

$$\left(\bigwedge_{j-i>1} \neg C(\tau_i, \tau_j) \right) \wedge \tau_0 \neq \mathbf{0} \wedge \tau_{2^n-1} \neq \mathbf{0} \wedge c(\mathbf{1})$$

forces an exponentially long 'chain', e.g.:

0			$2^n - 1$

- This allows us to encode computations of any polynomial-space-bounded Turing machine.

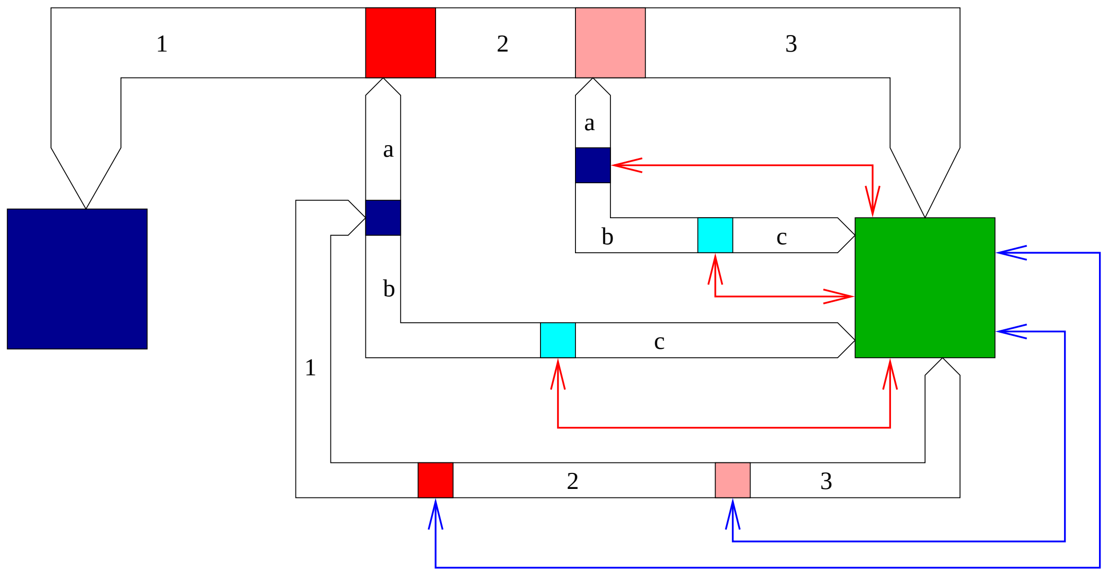
- As a result, we have:

$\mathcal{S}at(\mathcal{C}c^1, \text{RegC})$ is PSPACE-hard.

- A matching upper bound is available:

$\mathcal{S}at(\mathcal{S}4_u c^1, \text{RegC})$ is in PSPACE.

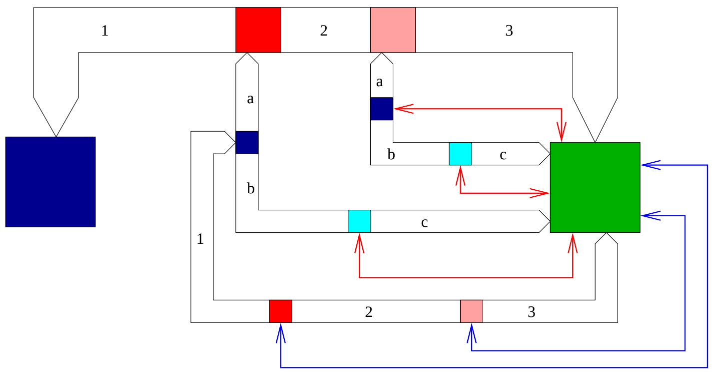
- What happens if we are allowed the full power of $\mathcal{C}c$?
- With just two occurrences of c, we can ‘enforce’ the following configuration:



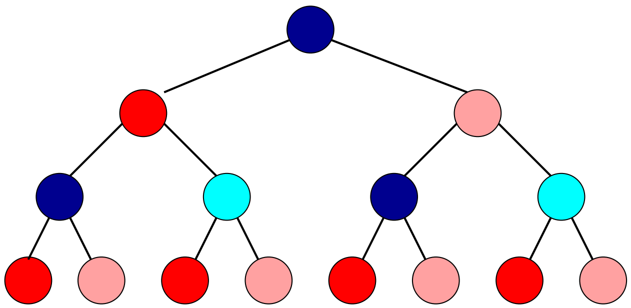
$$c(\text{Blue} + 1 + \text{Dark red} + 2 + \text{Light red} + 3 + \text{Green})$$

$$c(\text{Red} + a + \text{Dark blue} + b + \text{Light blue} + c + \text{Green})$$

- This structure can be viewed as a tree:



$\Rightarrow$



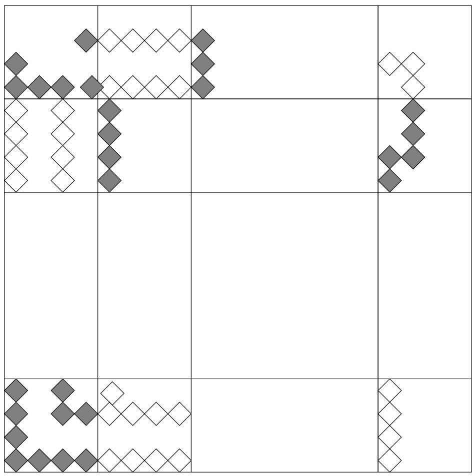
- This allows us to encode computations of any polynomial-space-bounded alternating Turing machine.
- As a result, we have:

$$\mathcal{S}at(\mathcal{C}c, \text{RegC}) \text{ is ExpTime-hard}$$

- A matching upper bound is available:

$$\mathcal{S}at(\mathcal{S}4_u c, \text{RegC}) \text{ is in ExpTime.}$$

- Actually, $\mathcal{C}c$-formulas can encode exponentially large grids.
- We take sequences of variables $r_n, \ldots, r_1$ and $s_n, \ldots, s_1$, using them to encode pairs of numbers (i, j) $(0 \leq i, j, < 2^n)$.
- And we use the number-coding tricks (familiar from $\mathcal{C}c$) to create a 'chess'-board pattern



with each square corresponding to a product

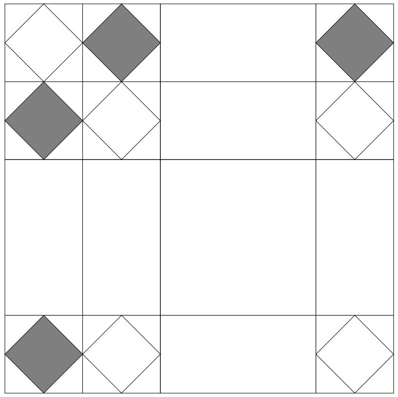
$$\pm r_n \cdot \ \cdots \ \cdot \pm r_1 \cdot \pm s_n \cdot \ \cdots \ \cdot \pm s_1.$$

and the grid connectivity represented by C.

- In the larger language $\mathcal{C}cc$, we can add formulas

$$c^{\leq 2^{n-1}}((r_0 \cdot s_0) + (-r_0 \cdot -s_0)) \qquad c^{\leq 2^{n-1}}((r_0 \cdot -s_0) + (-r_0 \cdot s_0)).$$

- This enforces connectedness of each of the 2^{n-1} black squares and each of the 2^{n-1} white squares:



- As a result, we have:

$$\mathcal{S}at(\mathcal{C}cc, \textsc{RegC}) \text{ is } \textsc{NExpTime}\text{-hard.}$$

- A matching upper bound is available:

$$\mathcal{S}at(\mathcal{S}4_u cc, \textsc{RegC}) \text{ is in } \textsc{NExpTime}.$$

- We mention in passing that, in the presence of the connectedness predicate, we can drop the predicate C in the languages $\mathcal{C}c$ and $\mathcal{C}cc$.
- Thus, $\mathcal{B}c$ is defined by the signature

$$\mathbf{0}, \quad \mathbf{1}, \quad +, \quad \cdot, \quad c$$

  and $\mathcal{B}cc$ is defined by the signature

$$\mathbf{0}, \quad \mathbf{1}, \quad +, \quad \cdot, \quad c^{\leq k}, \quad c^{\geq k}$$

  where $k \geq 0$.
- This reduces expressive power, but not complexity:

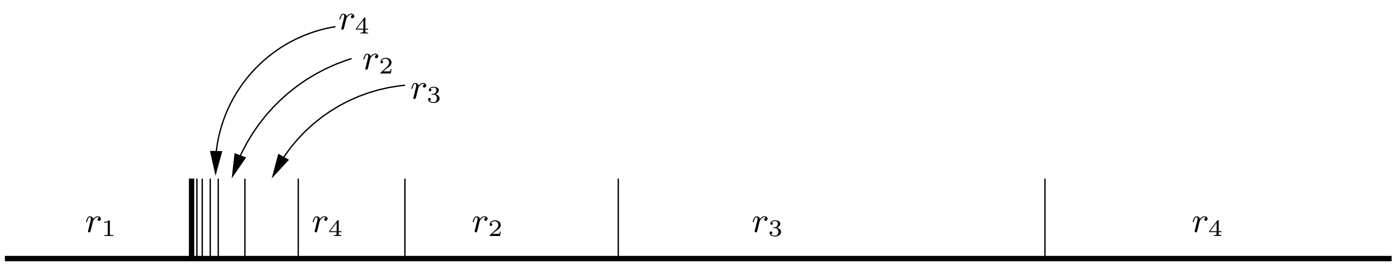
  $\mathcal{S}at(\mathcal{B}c, \textsc{RegC})$ is ExpTime-complete;
  $\mathcal{S}at(\mathcal{B}cc, \textsc{RegC})$ is NExpTime-complete.

- Finally, we consider what happens when the languages $\mathcal{RCC}$-8c, $\mathcal{C}c$ and $\mathcal{C}c$ are interpreted over low-dimensional Euclidean spaces.
- For the spaces $\mathbb{R}^n$, it is natural to consider the frames
  - $(\mathbb{R}^n, \boldsymbol{RC}(\mathbb{R}^n))$—the regular closed sets in $\mathbb{R}^n$;
  - $(\mathbb{R}^n, \boldsymbol{RCS}(\mathbb{R}^n))$—the reg. closed semi-algebraic sets in $\mathbb{R}^n$.
- Recall that the semi-algebraic sets count as 'tame':
  - They have finitely many components
  - They have the 'curve-selection' property

- Consider the $\mathcal{RCC}$-8c-formula

$$c(r_1) \wedge \bigwedge_{1 \le i < j \le 4} \mathsf{EC}(r_i, r_j).$$

- This formula is satisfiable over $(\mathbb{R}, \boldsymbol{RC}(\mathbb{R}))$, e.g. by



But it is not satisfiable over $(\mathbb{R}, \boldsymbol{RCS}(\mathbb{R}))$.

- Thus, $\mathcal{S}at(\mathcal{RCC}\text{-}8c, (\mathbb{R}, \boldsymbol{RC}(\mathbb{R}))) \neq \mathcal{S}at(\mathcal{RCC}\text{-}8c, (\mathbb{R}, \boldsymbol{RCS}(\mathbb{R})))$.
- We know that: $\mathcal{S}at(\mathcal{RCC}\text{-}8c, (\mathbb{R}, \boldsymbol{RCS}(\mathbb{R})))$ is NPTIME-complete; $\mathcal{S}at(\mathcal{RCC}\text{-}8c, (\mathbb{R}, \boldsymbol{RC}(\mathbb{R})))$ is NPTIME-hard and in PSPACE.
- Also: $\mathcal{S}at(\mathcal{C}c, (\mathbb{R}, \boldsymbol{RC}(\mathbb{R}))) \neq \mathcal{S}at(\mathcal{C}c, (\mathbb{R}, \boldsymbol{RCS}(\mathbb{R})))$, and both these problems are PSPACE-complete.
- However: $\mathcal{S}at(\mathcal{B}c, (\mathbb{R}, \boldsymbol{RC}(\mathbb{R}))) = \mathcal{S}at(\mathcal{C}c, (\mathbb{R}, \boldsymbol{RCS}(\mathbb{R})))$, and this probelm is NPTIME-complete.

- In $\mathbb{R}^2$, a rather different picture emerges:
- We know
  $\mathcal{S}at(\mathcal{RCC}\text{-}8c, (\mathbb{R}, \boldsymbol{RC}(\mathbb{R}^2))) = \mathcal{S}at(\mathcal{RCC}\text{-}8c, (\mathbb{R}, \boldsymbol{RCS}(\mathbb{R}^2)))$.
- The problem $\mathcal{S}at(\mathcal{RCC}\text{-}8c, (\mathbb{R}, D))$, where D is the set of disc-homeomorphs in the plane, is NPTIME-complete (Schaefer, Sedgwick and Štefankovič, 2003).
- It is then easy to show that $\mathcal{S}at(\mathcal{RCC}\text{-}8c, (\mathbb{R}, \boldsymbol{RCS}))$ is also NPTIME-complete.
- However, we have

$$\begin{aligned}\mathcal{S}at(\mathcal{B}c, (\mathbb{R}, \boldsymbol{RC}(\mathbb{R}^2))) &\neq \mathcal{S}at(\mathcal{B}c, (\mathbb{R}, \boldsymbol{RCS}(\mathbb{R}^2))) \\ \mathcal{S}at(\mathcal{C}c, (\mathbb{R}, \boldsymbol{RC}(\mathbb{R}^2))) &\neq \mathcal{S}at(\mathcal{C}c, (\mathbb{R}, \boldsymbol{RCS}(\mathbb{R}^2))).\end{aligned}$$

  The decidability of these problems is not known.

- Conceptual summary:
  - Topological langauges: $\mathcal{RCC}$-8, $\mathcal{B}$, $\mathcal{C}$, $\mathcal{S}4_u$ plus c, $c^{\leq k}$ $c^{\geq k}$
  - Topological frame classes: ALL, REGC, $\{(\mathbb{R}^n, \boldsymbol{RCS}(\mathbb{R}^n))\}$.
- Technical summary:

	REGC	$\boldsymbol{RC}(\mathbb{R})$	$\boldsymbol{RCS}(\mathbb{R})$	$\boldsymbol{RC}(\mathbb{R}^2)$	$\boldsymbol{RCS}(\mathbb{R}^2)$
$\mathcal{RCC}$-8c	NP	$\geq$NP $\leq$ PSPACE	NP	NP	NP
$\mathcal{RCC}$-8cc	NP	$\geq$NP $\leq$ PSPACE	$\geq$NP$\leq$ EXPTIME	$\geq$NP $\leq$ NEXPTIME	$\geq$NP $\leq$ NEXPTIME
$\mathcal{B}c$	EXPTIME	NP	NP	$\geq$ PSPACE	$\geq$ PSPACE
$\mathcal{C}c$	EXPTIME	PSPACE	PSPACE	$\geq$ EXPTIME	$\geq$ EXPTIME
$\mathcal{B}cc$	NEXPTIME	$\geq$ NP$\leq$ PSPACE	$\geq$ NP	$\geq$ PSPACE	$\geq$ PSPACE
$\mathcal{C}cc$	NEXPTIME	PSPACE	$\geq$ PSPACE	$\geq$ NEXPTIME	$\geq$ NEXPTIME

	ALL	$\mathbb{R}$	$S(\mathbb{R})$	$\mathbb{R}^2$	$S(\mathbb{R}^2)$
$\mathcal{S}4_u c$	EXPTIME	PSPACE	PSPACE	$\geq$ EXPTIME	$\geq$ EXPTIME
$\mathcal{S}4_u cc$	NEXPTIME	PSPACE	PSPACE	$\geq$ NEXPTIME	$\geq$ NEXPTIME