Model checking of CTL

Theorem

Let $M = (S, T, I, \mathrm{AP}, \ell)$ be a Kripke structure and $\varphi \in \mathrm{CTL}$ a formula.
The set $[\![\varphi]\!] = \{s \in S \mid M, s \models \varphi\}$ can be computed in time $\mathcal{O}(|M| \cdot |\varphi|)$.
Hence, the model checking problem $M \models_\exists \varphi$ is decidable in time $\mathcal{O}(|M| \cdot |\varphi|)$.

Proof:

Compute $[\![\varphi]\!]$ by induction on the formula.

The set $[\![\varphi]\!]$ is represented by a boolean array: $L[s] = \top$ if $s \in [\![\varphi]\!]$.

For each $t \in S$, the set $T^{-1}(t)$ is represented as a *list*.

T^{-1} is an array of lists, its size is $|S| + |T|$.

`for all` $t \in S$ `do for all` $s \in T^{-1}(t)$ `do ... od` takes time $\mathcal{O}(|S| + |T|)$.

Model checking of CTL

Definition: function semantics(φ) returns boolean array L

case $\varphi = p \in \mathrm{AP}$
for all $s \in S$ do $L[s] := (p \in \ell(s))$ od — $\mathcal{O}(|S|)$

case $\varphi = \neg\varphi_1$
$L_1 :=$ semantics(φ_1)
for all $s \in S$ do $L[s] := \neg L_1[s]$ od — $\mathcal{O}(|S|)$

case $\varphi = \varphi_1 \vee \varphi_2$
$L_1 :=$ semantics(φ_1); $L_2 :=$ semantics(φ_2)
for all $s \in S$ do $L[s] := L_1[s] \vee L_2[s]$ od — $\mathcal{O}(|S|)$

case $\varphi = EX\varphi_1$
$L_1 :=$ semantics(φ_1)
for all $s \in S$ do $L[s] := \bot$ od — $\mathcal{O}(|S|)$
for all $t \in S$ do if $L_1[t]$ then for all $s \in T^{-1}(t)$ do $L[s] := \top$ — $\mathcal{O}(|S| + |T|)$

case $\varphi = AX\varphi_1$
$L_1 :=$ semantics(φ_1)
for all $s \in S$ do $L[s] := \top$ od — $\mathcal{O}(|S|)$
for all $t \in S$ do if $\neg L_1[t]$ then for all $s \in T^{-1}(t)$ do $L[s] := \bot$ — $\mathcal{O}(|S| + |T|)$

Model checking of CTL

Definition: function semantics(φ) returns boolean array L

case $\varphi = \mathsf{E}\,\varphi_1\,\mathsf{U}\,\varphi_2$ — $\mathcal{O}(|S| + |T|)$

$L_1 :=$ semantics(φ_1); $L_2 :=$ semantics(φ_2)
for all $s \in S$ do — $\mathcal{O}(|S|)$
  $L[s] := L_2[s]$
  if $L_2[s]$ then Todo.add(s) // Todo is implemented with a stack
while $\mathrm{Todo} \neq \emptyset$ do — $|S|$ times
Invariant 1: $[\![\varphi_2]\!] \cup \mathrm{Todo} \subseteq L \subseteq [\![\mathsf{E}\,\varphi_1\,\mathsf{U}\,\varphi_2]\!]$
  $t := \mathrm{Todo}$.remove() — $\mathcal{O}(1)$
  for all $s \in T^{-1}(t)$ do — $|T|$ times
    if $L_1[s] \wedge \neg L[s]$ then Todo.add(s); $L[s] := \top$ — $\mathcal{O}(1)$
od

Model checking of CTL

Definition: function semantics(φ) returns boolean array L

case $\varphi = \mathsf{A}\,\varphi_1\,\mathsf{U}\,\varphi_2$ $\mathcal{O}(|S| + |T|)$
  $L_1 :=$ semantics(φ_1); $L_2 :=$ semantics(φ_2)
  for all $s \in S$ do $\mathcal{O}(|S|)$
    $L[s] := L_2[s]$
    if $L_2[s]$ then Todo.add(s) // Todo is implemented with a stack
  for all $s \in S$ do $d[s] := 0$ $\mathcal{O}(|S|)$
  for all $t \in S$ do for all $s \in T^{-1}(t)$ do $d[s] := d[s] + 1$ $\mathcal{O}(|S| + |T|)$
  while $\mathrm{Todo} \neq \emptyset$ do $|S|$ times
  Invariant 1: $\forall s \in S,\ |T(s)| - d[s] = |T(s) \cap (L \setminus \mathrm{Todo})|$
  Invariant 2: $[\![\varphi_2]\!] \cup \mathrm{Todo} \subseteq L \subseteq [\![\mathsf{A}\,\varphi_1\,\mathsf{U}\,\varphi_2]\!]$
    $t := \mathrm{Todo}$.remove() $\mathcal{O}(1)$
    for all $s \in T^{-1}(t)$ do $|T|$ times
      $d[s] := d[s] - 1$ $\mathcal{O}(1)$
      if $L_1[s] \wedge \neg L[s] \wedge d[s] = 0$ then Todo.add(s); $L[s] := \top$ $\mathcal{O}(1)$
  od