Generic Infinite Traces  
and  
Path-Based Coalgebraic Temporal Logics

Corina Cîrstea

School of Electronics and Computer Science  
University of Southampton
Overview

• several known path-based temporal specification logics:
  • CTL* on transition systems
  • PCTL on probabilistic transition systems
• similarities not sufficiently understood/exploited

Goals:
• find a unifying pattern (need infinite computation paths)
  • existing general theory of finite traces [Hasuo et. al.]
  • existing definition of infinite traces for $T = \mathcal{P}$ [Jacobs ’04]
• automatically derive new path-based temporal logics
Restricted Transition Systems

- restricted transition systems are $\mathcal{P}^+$-coalgebras

\((\mathcal{P}^+ (S) = \text{set of non-empty subsets of } S)\)

Example

Some computation paths from $s_0$:

- $s_0 \rightarrow s_1 \rightarrow s_1 \ldots$
- $s_0 \rightarrow s_1 \rightarrow s_2 \rightarrow s_0 \rightarrow s_1 \rightarrow s_2 \ldots$
- $s_0 \rightarrow s_1 \rightarrow s_3 \rightarrow s_3 \ldots$

- to each state, one associates a set of computation paths
The Logic CTL*

- **path formulas:**  \( \varphi ::= \phi \mid \neg \varphi \mid \varphi \land \varphi \mid X\varphi \mid F\varphi \mid G\varphi \mid \varphi U \varphi \)

- **state formulas:**  \( \phi ::= \tt \mid \mathit{p} \mid \neg \phi \mid \phi \land \phi \mid E\varphi \mid A\varphi \)

- **E** and **A** similar to \( \Diamond \) and \( \Box \) modalities . . .

**Example**

\[
A \ F (\mathit{try} U \mathit{succ})
\]
Probabilistic Transition Systems

- probabilistic transition systems are $D$-coalgebras
  \[ D(S) = \text{set of probability distributions over } S \]

Example

Some computation paths from $s_0$:

- $s_0 \rightarrow s_1 \rightarrow s_1 \ldots$
- $s_0 \rightarrow s_1 \rightarrow s_2 \rightarrow s_0 \rightarrow s_1 \rightarrow s_2 \ldots$
- $s_0 \rightarrow s_1 \rightarrow s_3 \rightarrow s_3 \ldots$

- to each state, one associates a probability measure on the computation paths from that state
The Logic PCTL

- **path formulas:** $\varphi ::= X\phi \mid \phi U^{\leq t} \phi \quad t \in \{0, 1, \ldots\} \cup \{\infty\}$

- **state formulas:** $\phi ::= tt \mid p \mid \neg \phi \mid \phi \land \phi \mid [\varphi]_{\geq q} \mid [\varphi]_{>q}$

**Example**

![Diagram](image)

- $[tt U^{\leq 3} fail] < 0.1$
- $[(try U succ)] \geq 1$
More Examples

• (restricted) labelled transition systems (LTSs) are \( \mathcal{P}^+(A \times \text{Id}) \)-coalgebras

• generative probabilistic transition systems (GPTSs) are \( \mathcal{D}(A \times \text{Id}) \)-coalgebras

For both LTSs and GPTSs, computation paths have the form

\[
s_0 \xrightarrow{a_0} s_1 \xrightarrow{a_1} s_2 \xrightarrow{a_2} \ldots
\]

whereas infinite computation traces have the form

\[
a_0 a_1 a_2 \ldots
\]

What LTSs and GPTSs have in common is the inner part of the signature functor: \( A \times \text{Id} \).
The General Setting

Similarly to [Hasuo et. al.], we focus on $T \circ F$-coalgebras, where:

- **strong monad** $T : C \to C$ describes the computation type
  
  e.g. $\mathcal{P}^+$, $\mathcal{D}$

- functor $F : C \to C$ describes the transition type
  
  - require final sequence of $F$ to stabilise at $\omega$
    
    e.g. $\text{Id}$, $A \times \text{Id}$, $1 + A \times \text{Id}$

- distributive law $\lambda : F \circ T \Rightarrow T \circ F$ (compatible with monad structure) is fixed
Towards Infinite Traces

- the possible infinite traces for both LTSs and GPTSs are elements of $A^\omega$ (the final $A \times -$coalgebra):

$$\begin{align*}
1 & \xleftarrow{} A & A \xleftarrow{} A \times \bar{A} & \xleftarrow{} \cdots
\end{align*}$$

- for an LTS/GPTS $(S, \gamma)$, the actual infinite traces should be structured according to the computation type:

$$tr_{\gamma} : S \rightarrow P^+(A^\omega) \quad \text{or} \quad tr_{\gamma} : S \rightarrow D(A^\omega)$$
Defining the Infinite Trace Map (for LTSs)

Fix an LTS $\gamma : S \rightarrow P^+(A \times S)$.

Define $tr_\gamma : S \rightarrow P^+(A^\omega)$ from its finite approximants $\gamma_i$.

For existence of $tr_\gamma$, we need:

- $\gamma_i$’s define cone
- $P^+(A^\omega)$ weakly limiting
Defining the Approximants (for LTSs)

\[ \gamma : S \rightarrow \mathcal{P}^+(S) \]

\[ \gamma(s_0) = \{(a, s_1)\} \]
\[ \gamma(s_1) = \{(a, s_2), (b, s_3), (c, s_1)\} \]
\[ \gamma(s_2) = \{(b, s_0)\} \]
\[ \gamma(s_3) = \{(c, s_3)\} \]

- one application of \( \gamma \) gives

\[ \gamma_1(s_1) = \{a, b, c\} \]

- two applications of \( \gamma \) followed by some “flattenning” (use of distributive law) give

\[ \gamma_2(s_1) = \{ab, bc, ca, cb, cc\} \]

- ...
A Problem . . . and its Solution

\[ S \xrightarrow{\gamma} \cdots \xrightarrow{\gamma_0, \gamma_1, \gamma_2} \cdots \xrightarrow{\text{tr}_{\gamma}} \mathcal{P}^+(A^\omega) \]

\[ \mathcal{P}^+(1) \leftarrow \mathcal{P}^+(A) \leftarrow \mathcal{P}^+(A \times A) \leftarrow \cdots \]

- in general, there are several choices for the infinite trace map . . .
- . . . but there is a canonical (maximal) one, assuming:
  - dcpo \( \sqsubseteq \) on \( S \rightarrow \mathcal{P}^+(Z) \)
  - mediating maps form directed set
- the trace map can be defined for a general coalgebraic type \( T \circ F \)
  (subject to reasonable constraints)
From Infinite Traces to Infinite Executions

- view $\mathcal{P}^+(A \times \_)$-coalgebra: as $\mathcal{P}^+(S \times A \times \_)$:

- obtain an infinite execution map $\text{exec}_\gamma : S \to (S \times A)^\omega$ as the infinite trace map of the new coalgebra!!
“Infinite” Executions: Examples

Take $T = \mathcal{P}^+$. 

- $F = _-$ (restricted TSs):
  
  $s_0 \ s_1 \ s_2 \ldots$

- $F = A \times _-$ (restricted LTSs):
  
  $s_0 \ a_1 \ s_1 \ a_2 \ s_2 \ldots$

- $F = 1 + A \times _-$ (LTSs):
  
  $s_0 \ a_1 \ s_1 \ a_2 \ s_2 \ldots$  or  $s_0 \ a_1 \ s_1 \ldots \ s_n$
The Case of Probabilistic Systems

Example

- working with $T = D$ over sets does not work:
  - probability measures needed to deal with \textit{uncountably many} traces
  \Rightarrow need to work with $T = \mathcal{G}$ (the \textit{Giry monad}) over measurable spaces

- resulting infinite trace map takes states to probability measures over infinite traces
Coalgebra Structure on Infinite Executions

Fix a $\mathcal{P}^+(A \times \_)$-coalgebra $(S, \gamma)$.

The possible infinite executions have $S \times (A \times \_)$-coalgebra structure.

Hence, one can extract from each infinite execution

- the first state,
- an $A \times \_$-observation.
Towards Coalgebraic Path-Based Temporal Logics

- coalgebraic types come equipped with modal languages
- e.g. for $T = \mathcal{P}^+$, the language has modal operators $\Box$ and $\Diamond$:
  - $s \models \Box \phi$ iff $s' \models \phi$ for all $s'$ s.t. $s \rightarrow s'$
  - $s \models \Diamond \phi$ iff $s' \models \phi$ for some $s'$ s.t. $s \rightarrow s'$
- e.g. for $F = A \times -$, the language has modal operators $a$ and $X$:
  - $s \models a$ iff $s \rightarrow (a, s')$
  - $s \models X\phi$ iff $s \rightarrow (a, s')$ and $s' \models \phi$
- our coalgebras have type $T \circ F$, so we make use of the above . . .
  - . . . but with a non-standard interpretation of $\Box$ and $\Diamond$!
Path-Based Fixpoint Logics (for TSs)

\[ T = \mathcal{P}^+ \text{ with monotone } \Box, \Diamond \]

\[ F = \text{Id with monotone } X \]

\[ \varphi ::= \texttt{tt} | \texttt{ff} | p^F | \phi | \varphi \land \varphi | \varphi \lor \varphi | X\varphi | \mu p^F.\varphi | \nu p^F.\varphi \]

\[ \phi ::= \texttt{tt} | \texttt{ff} | p | \phi \land \phi | \phi \lor \phi | \Box \varphi | \Diamond \varphi \]

Given \( T \circ F\)-coalgebra \((S, \gamma)\) and suitable valuations (for \( p^F \) and \( p \)), interpret

- **path formulas** \( \varphi \) as sets of paths
  - use \( S \times F\)-coalgebra structure on \( S^\omega \) to interpret \( \phi \) and \( X\varphi \)

- **state formulas** \( \phi \) as sets of states
  - use infinite execution map \( \text{exec}_\gamma : S \to \mathcal{P}^+(S^\omega) \) to interpret \( \Box \varphi \), \( \Diamond \varphi \)
General Path-Based Fixpoint Logics

Fix

- base category $C$ with $U : C \to \text{Set}$
- functor $P : C \to \text{Set}^{\text{op}}$ specifying admissible predicates
  - assume $PC \subseteq \mathcal{P}UC$ is a complete lattice
- functors $T$ and $F$ with monotone modal operators $\Lambda$ and $\Lambda_F$, resp.

Definition (Path-Based Fixpoint Language Syntax)

$$
\varphi ::= \text{tt} | \text{ff} | p^F | \phi | \varphi \land \varphi | \varphi \lor \varphi | [\lambda_F]\varphi | \mu p^F.\varphi | \nu p^F.\varphi
$$

$$
\phi ::= \text{tt} | \text{ff} | p | \phi \land \phi | \phi \lor \phi | [\lambda]\varphi
$$

- semantics as expected …
Recovering (negation-free) CTL*

Define:

- $X\varphi ::= X\varphi$
- $F\varphi ::= \mu X.(\varphi \lor XX)$
- $G\varphi ::= \nu X.(\varphi \land XX)$
- $\varphi U \psi ::= \mu X.(\psi \lor (\varphi \land XX))$
  
  ...  

- $A\varphi ::= \square \varphi$
- $E\varphi ::= \lozenge \varphi$
How About LTSs?

\( T = P^+ \) with modal operators \( \square, \Diamond \)

\( F = A \times \text{Id} \) with modal operators \( a \ (a \in A), \mathbf{X} \)

\[
\Rightarrow \quad \varphi ::= \text{tt} \mid \text{ff} \mid p^F \mid \phi \mid \varphi \land \varphi \mid \varphi \lor \varphi \mid a \mid \mathbf{X} \varphi \mid \mu p^F . \varphi \mid \nu p^F . \varphi
\]

\[
\phi ::= \text{tt} \mid \text{ff} \mid p \mid \phi \land \phi \mid \phi \lor \phi \mid \square \varphi \mid \Diamond \varphi
\]

- CTL* operators defined as before!
- can refer to the next label along a path:
  - natural encoding of “\( a \) occurs along every path” as
    \[
    \square F a ::= \square \mu X.( a \lor XX )
    \]
  - compare above to
    \[
    \mu X.( \langle \_ \rangle \text{tt} \land \lnot a \mathbf{X} )
    \]
Logics with (Existential) Until Operators

- assume $PC \subseteq PUC$ is a $\sigma$-algebra
- replace fixpoint operators with Until operators $\mathbf{U}_L$
  - $L \subseteq \Lambda_F$ finite set of (disjunction-preserving) predicate liftings
- semantics defined by

\[
\mathcal{L} \phi \mathbf{U}_L \psi \mathcal{M} = \bigcup_{i \in \omega} \mathcal{L} \phi \mathbf{U}_L^{\leq i} \psi
\]

where

\[
\mathcal{L} \phi \mathbf{U}_L^{\leq 0} \psi \mathcal{M} := \psi
\]

\[
\mathcal{L} \phi \mathbf{U}_L^{\leq i+1} \psi \mathcal{M} := \psi \lor (\phi \land \bigvee_{\lambda F \in L} [\lambda F](\phi \mathbf{U}_L^{\leq i} \psi))
\]
Recovering PCTL as a Fragment

\[ T = D, \quad F = \text{Id} \]

\[ \Lambda = \{L_q\}, \quad \Lambda_f = \{X\} \]

\[ \varphi ::= \text{tt} \mid \text{ff} \mid \phi \mid \varphi \land \varphi \mid \varphi \lor \varphi \mid X\varphi \mid \varphi U_X \varphi \]

\[ \phi ::= \text{tt} \mid p \mid \neg \phi \mid \phi \land \phi \mid L_q \varphi \]

Define:

- \( X\varphi ::= X\varphi \)
- \( \varphi U \psi ::= \varphi U_X \psi \)
- \( [\varphi]_{\geq q} ::= L_q \varphi \)
Future Work

- other computational monads
  - e.g. the finite multiset monad and graded temporal logics?
- investigate linear fragments of path-based temporal logics
  - automata-based model-checking techniques (parameterised by computation type)