Temporal Logics

CS 294-3: Techniques for Automated Deduction
(Prof George C. Necula)

Arindam Chakrabarti

Outline

- Reactive systems, Kripke structures
- What is a temporal logic ?
- Logics in verification: LTL
- Logics in verification: CTL
- Fixpoint characterizations: Algorithms

Outline

- Reactive systems, Kripke structures
- What is a temporal logic ?
- Logics in verification: LTL
- Logics in verification: CTL
- Fixpoint characterizations: Algorithms

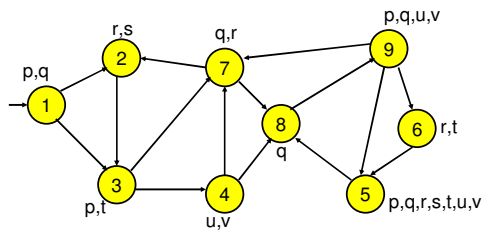
Reactive systems

- Transformational systems eg. compiler
  - Correctness requirements can be expressed in terms of initial and final states
  - Hoare logic
- Reactive systems eg. OS, network protocols
  - Correctness criteria depends on context
  - Temporal logic useful to express correctness properties

Kripke structures

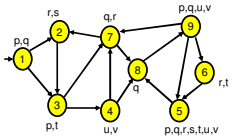
- Model for representing reactive systems
- *AP:* set of atomic propositions
- *K:* Kripke structure over $AP = (S, S_0, T, L)$
- *S:* set of states (need not be finite)
- S_0: set of initial states
- *T:* transition relation $(T \subseteq S \times S)$
- *L:* labelling function $(L : S \to 2^{AP})$

Example



Example

- AP = {p,q,r,s,t,u,v}
- S = {1,2,3,4,5,6,7,8,9}
- S_0 = {1}
- T = {(1,2), (1,3),(2,3), …}
- L = {(1,{p,q}),(2,{r,s}), …}



Outline

- Reactive systems, Kripke structures
- What is a temporal logic ?
- Logics in verification: LTL
- Logics in verification: CTL
- Fixpoint characterizations: Algorithms

What is a temporal logic ?

- Logic with *temporal* operators (operators that talk about time)
  - Eg. Tense Logic (A. N. Prior, 1957)
    - P "It has at some time been the case that …"
    - F "It will at some time be the case that …"
    - H "It has always been the case that …"
    - G "It will always be the case that …"

Duality relationships

- $Pp \equiv \neg H \neg p$
- $Fp \equiv \neg G \neg p$

Relationship between past and future

- $p \Rightarrow HFp$
  - "What is, has always been going to be"
- $p \Rightarrow GPp$
  - "What is, will always have been"
- $H(p \Rightarrow q) \Rightarrow (Hp \Rightarrow Hq)$
  - "Whatever always follows from what always has been, always has been"
- $G(p \Rightarrow q) \Rightarrow (Gp \Rightarrow Gq)$
  - "Whatever always follows from what always will be, always will be"

Axioms

- $Gp \Rightarrow Fp$
  - "What will always be, will be"
- $G(p \Rightarrow q) \Rightarrow (Gp \Rightarrow Gq)$
  - "If p will always imply q, then if p will always be the case, so will q"
- $Fp \Rightarrow FFp$
  - "If it will be the case that p, it will be — in between — that it will be"
- $\neg Fp \Rightarrow F\neg Fp$
  - "If it will never be that p then it will be that it will never be that p"

Inference rules

- $RG : \frac{p}{Gp}$
- $RH : \frac{p}{Hp}$

Kinds of temporal logics

- Propositional or first order
- Global or compositional
- Branching time or linear time
- Points or intervals
- Discrete time or continuous time
- Past or future

Outline

- Reactive systems, Kripke structures
- What is a temporal logic ?
- Logics in verification: LTL
- Logics in verification: CTL
- Fixpoint characterizations: Algorithms

Logics in verification

- Linear Temporal Logic (LTL)
  - The underlying structure of time is a totally ordered set isomorphic to $(\mathbb{N}, <)$
- Computational Tree Logic (CTL)
  - The underlying structure of time is an infinite tree.

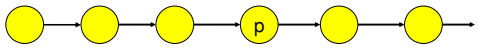
(Propositional) LTL

Atomic propositions, propositional connectives, and the following temporal operators:

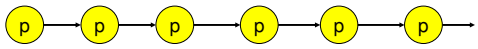
- F p
  - "sometime in the Future, p holds"
- G p
  - "at all times (Globally), p holds"
- X p
  - "in the neXt state, p holds"
- p U q
  - "p holds Until q holds"
- P W q
  - "p holds Waiting for q"

Informal semantics

- F p

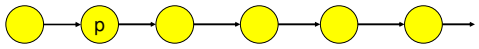


- G p

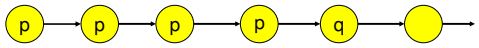


Informal semantics

- X p



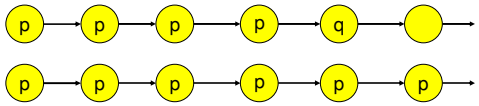
- p U q



q has to be eventually true !

Informal semantics

- p W q



q may never be true !

Satisfiability, Validity

- A linear time structure is an infinite totally ordered set of labelled states isomorphic to $(\mathbb{N}, <)$.
- An LTL formula ϕ is satisfiable if there exists a linear time structure M s.t. $M \models \phi$.
- An LTL formula ϕ is valid if for all linear time structures M, $M \models \phi$.

Additional operators

- p B q
  - "p Before q: If q ever happens in the future, it is strictly preceded by an occurrence of p"
- F^∞ p
  - "p holds infinitely often" (but may also fail to hold infinitely often)
- G^∞ p
  - "p holds almost always" (and fails to hold only finitely many times)

Duality and other relationships

- $\models G\, p \equiv \neg F\, \neg p$
- $\models X\, \neg p \equiv \neg\, X\, p$
- $\models G^\infty\, p \equiv \neg\, F^\infty\, \neg p$
- $\models F^\infty\, p \equiv GF\, p$
- $\models p\, B\, q \equiv \neg((\neg p)\, U\, q)$

Other relationships

- $\models p \Rightarrow F\, p$
- $\models G\, p \Rightarrow p$
- $\models X\, p \Rightarrow F\, p$
- $\models G\, p \Rightarrow X\, p$
- $\models G\, p \Rightarrow F\, p$
- $\models G\, p \Rightarrow XG\, p$
- $\models p\, U\, q \Rightarrow F\, q$
- $\models \neg(p\, W\, q \Rightarrow F\, q)$
- $\models G^\infty\, p \Rightarrow F^\infty\, p$

Outline

- Reactive systems, Kripke structures
- What is a temporal logic ?
- Logics in verification: LTL
- Logics in verification: CTL
- Fixpoint characterizations: Algorithms

(Propositional) CTL

- Atomic propositions, propositional connectives, *path quantifiers* (∀All, ∃Exists), and *temporal operators*: X, F, G, U, W.

CTL syntax

- ϕ_s :: p | $\phi_s \vee \phi_s$ | $\phi_s \wedge \phi_s$ | $\neg \phi_s$ | E ϕ_p | A ϕ_p
- ϕ_p :: X ϕ_s | ϕ_s U ϕ_s
- ϕ_s are called *state formulae.* They must be interpreted at a state. eg. p, p ∨ q, AX p, E(p U q).
- ϕ_p are called *path formulae.* They must be interpreted over a path (infinite sequence of states). eg. X p, p U q.

CTL semantics

- K, s ⊨ p iff state s is labelled with proposition p in Kripke structure K.
- K, s ⊨ $\phi_s^1 \vee \phi_s^2$ iff K, s ⊨ ϕ_s^1 or K, s ⊨ ϕ_s^2
- K, s ⊨ $\phi_s^1 \wedge \phi_s^2$ iff K, s ⊨ ϕ_s^1 and K, s ⊨ ϕ_s^2
- K, s ⊨ $\neg \phi_s$ iff ¬(K, s ⊨ ϕ_s)
- K, s ⊨ E ϕ_p iff ∃ path p = s, s', s'', … in K, K, p ⊨ ϕ_p.
- K, s ⊨ A ϕ_p iff ∀ path p = s, s', s'', … in K, K, p ⊨ ϕ_p.

CTL semantics

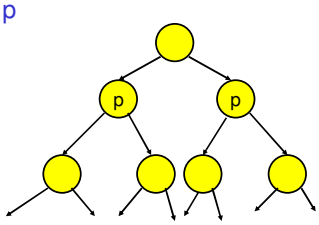
- K, p ⊨ ϕ_s^1 U ϕ_s^2 iff ∃ i.[K, p^i ⊨ ϕ_s^2 and ∀ j.(j < i ⇒ K, p^j ⊨ ϕ_s^1)], where $p^i = s_i, s_{i+1}, \ldots$ when p = $s_0, s_1, s_2, \ldots$
- K, p ⊨ X ϕ_s iff K, p^1 ⊨ ϕ_s.

CTL operators

- In other words, no two temporal operators may occur without being separated by path quantifiers.
- eg. XX p is an LTL formula, but not allowed in CTL. The logic CTL^* allows such formulae; CTL^* is a superset of both CTL and LTL.
- Thus, the CTL operators are:
  - AX, EX
  - AF, EF
  - AG, EG
  - AU, EU
  - AW, EW

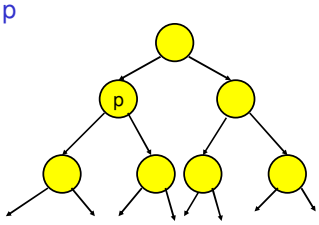
Informal semantics

- AX p



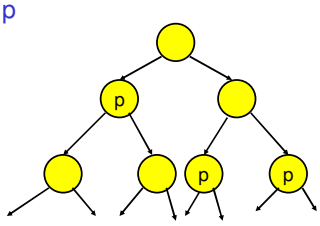
Informal semantics

- EX p



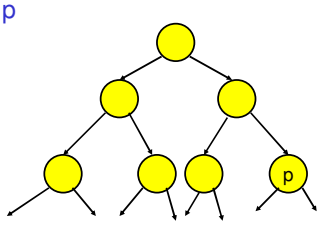
Informal semantics

- AF p



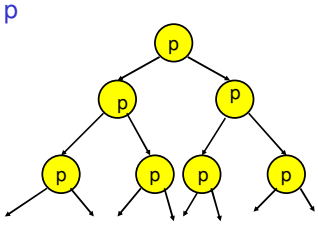
Informal semantics

- EF p



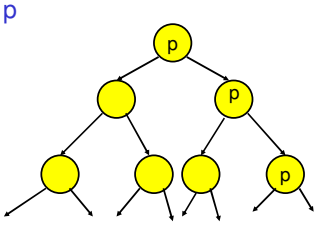
Informal semantics

- AG p



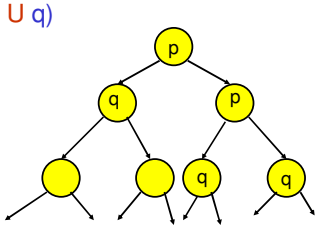
Informal semantics

- EG p



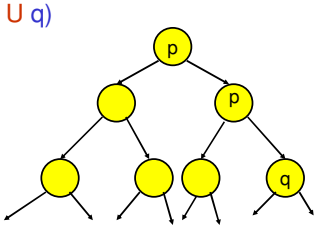
Informal semantics

- A(p U q)



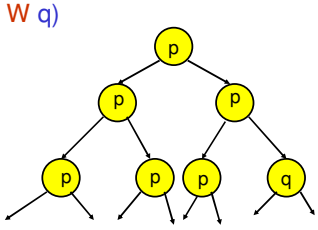
Informal semantics

- E(p U q)



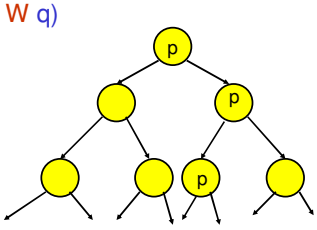
Informal semantics

- A(p W q)



Informal semantics

- E(p W q)



Examples

- EF (Start $\wedge$ $\neg$Ready)
  - "It is possible to get to a state where Start holds and Ready does not hold"
- AG(EF Restart)
  - "From any state it is possible to get to the Restart state"
- AG(AF DeviceEnabled)
  - "DeviceEnabled holds infinitely often on every path"
- AG(Req $\Rightarrow$ AF Ack)
  - "If a request occurs, it will be eventually acknowledged"

EX, EG, EU

- AX p = $\neg$EX $\neg$p
- EF p = E (TRUE U p)
- AG p = $\neg$ EF ($\neg$p)
- AF p = $\neg$ EG ($\neg$p)
- A (p U q) = $\neg$E ($\neg$q U ($\neg$p $\wedge$ $\neg$q) $\wedge$ $\neg$EG $\neg$q)
- A(p W q) = $\neg$E($\neg$ p U $\neg$q)
- E(p W q) = $\neg$A($\neg$ p U $\neg$q)

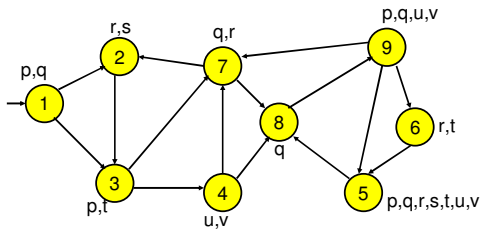
Other considerations

- Fairness
- Existential, Universal fragments

Outline

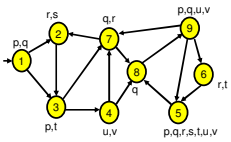
- Reactive systems, Kripke structures
- What is a temporal logic ?
- Logics in verification: LTL
- Logics in verification: CTL
- Fixpoint characterizations: Algorithms

A formula represents a set



A formula represents a set

$$[\![p]\!] = \{1,3,5,9\}$$
$$[\![q]\!] = \{1,5,7,8,9\}$$
$$[\![r]\!] = \{2,5,6,7\}$$
$$[\![p \vee q]\!] = [\![p]\!] \cup [\![q]\!]$$
$$[\![EXp]\!] = \{1,2,6,8,9\}$$
$$[\![AXp]\!] = \{2,6,8\}$$



Fixpoint formulae represent algorithms

lfp Z [p ∨ AX Z]

1. Start with $Z_0 = \emptyset$.
2. while $Z_i \neq Z_{i-1}$
3. {
4. $Z_{i+1} = p \vee AX\ Z_i$
5. i++;
6. }
7. return Z_i

Fixpoint characterizations of CTL operators

- A(p U q) = lfp Z [q ∨ (p ∧ AX Z)]
- E(p U q) = lfp Z [q ∨ (p ∧ EX Z)]
- AF p = lfp Z [p ∨ AX Z]
- EF p = lfp Z [p ∨ EX Z]
- AG p = gfp Z [p ∧ AX Z]
- EG p = gfp Z [p ∧ EX Z]

References

- *Model Checking.* E. M. Clarke, O. Grumberg, D. A. Peled. MIT Press.
- *Temporal and Modal Logic.* E Allen Emerson. Handbook of Th. Comp. Sci. Vol B. Elsevier Science.
- *Computer-Aided Verification.* R. Alur and T. A. Henzinger.

Thanks for listening !

Questions ?