FMUOS

1. Uvod

- formalne metode: matematički zasnovane tehnike za specificiranje i verifikaciju
- formalne metode: specifikacija, sinteza, verifikacija
- formalne metode su **statičke** → ne pokreće se ono što se analizira (za razliku od ispitivanja koje je **dinamičko**)
- glavni elementi:
  - model sustava
  - specifikacija
  - verifikacija (ulaz su model i specifikacija a izlaz T ili F)
- lagane metode: provjeravaju se samo dijelovi sustava
  - provjera samo nekih značajki npr.: sigurnost, životnost, nepristranost, odsutnost zastoja
  - primjer: **provjera modela**
- **specifikacija:**
  - ASM (stroj s konačnim brojem stanja) → konačan skup pravila
  - Z metoda: skup schema (State, Operation, Observation)
- **sinteza:**
  - compiler, SDL
- **verifikacija:**
  - za razliku od testiranja zbilja dokazuje odsutnost pogreški
  - metode:
    - provjera ekvivalentnosti
    - **provjera modela**
      - napravi model (verilog, NuSMV), specifikaciju prikaži vremenskom logikom(CTL), verifikacija je automatizirana
      - gledamo da li model sustava **logički zadovoljava** specifikaciju
    - dokazivanje teorema
    - provjera tvrdnje
- metode:
  - vrlo lagane (neispravne, nekompletne)
  - srednje teške (ispravne, nekompletne)
  - teške (ispravne, kompletne)

2.1. Logika

- propozicijska i predikatna logika, vremenska logika
- interpretacija: pridruživanje istinitosti atomima, npr p = T, q = F, f = T
  - interpretacija je **model** ako su za nju sve formule nekog formalnog sustava istinite (može se sve formule spojit u jednu pa je onda interpretacija model ako je za nju ta formula istinita)
  - **zadovoljivi** skup formula ima barem jedan model.
  - Formula je **logička posljedica** skupa formula ako je svaki model od skupa formula ujedno i model formule

- semantika: pridruživanje istinitosti formuli → interpretacija i evaluacija
- bitno pravilo: A → B = !A v B
- formalan logički sustav je dvojka (konačan skup ispravnih formula (wff), konačan skup pravila zaključivanja)
  - modus ponens: ako P = T i P → Q = T onda Q = T
  - modus tolens: ako Q = F i P → Q = T onda P = F
- **teorem:** formula koja već postoji ili se može izvesti
  - skup formula je **konzistentan** ako ne sadrži kontradiktorne teoreme
- podjela po odredivosti: odrediv, poluodrediv, neodrediv formalni sustav
- ako je svaki teorem ujedno i logička posljedica onda je formalan sustav **ispravan**
- ako se svaka logička posljedica može dobiti teoremom onda je formalan sustav **kompletan**
- normalni oblici formula: **DNF** i **CNF**
- **teorem dedukcije**: B je logička posljedica od A ako je (A ^ !B) nezadovoljiva
  - logično, jer (A → B) = tautologija ako je !(A → B) nezadovoljivo, tj. ako je (A ^ !B) nezadovoljivo
- **SAT problem** (problem zadovoljivosti → da li je formula zadovoljiva?)
  - težak
  - za DNF je polinomijalne složenosti, za CNF je NP kompletno (dakle lošije)
- **Predikatna logika**
  - atomi, predikati, kvantifikatori
  - V ide uz →, E ide uz ^
  - **poluodrediva**, ispravna i kompletna

2.2. Vremenska logika(CTL)

- postoje: **propozicijska**, predikatna
- postoje: **globalna**, linearna, **grananje vremena**, **diskretna**/kontinuirana, prošla/**buduća**
- uzimamo: **propozicijska, globalna, grananje, buduće vrijeme**
- **Model implementacije je Kripke struktura (S, R, L):**
  - S je skup svih stanja
  - R je skup svih relacija
  - L je skup oznaka
- operatori:
  - **X** f (f vrijedi za iduće stanje),
  - **F** f (f vrijedi za ovo ili neko buduće stanje),
  - **G** f (f vrijedi za ovo i svako buduće stanje),
  - f **U** g (u budućnosti ili sada postoji stanje gdje vrijedi g, a do tada vrijedi f)
- kvantifikatori: A, E
- obilježja:
  - sigurnosti: nešto loše se neće dogoditi (AG)
  - životnosti: nešto dobro će se konačno (u nekom trenutku) dogoditi (AF)
- fairness:
  - kažemo što želimo da vrijedi beskonačno često → samo se takvi putovi gledaju

5.1. LTL (Linear Temporal Logic)

- Neke stvari se mogu izraziti u CTL ali ne u LTL i obrnuto (dakle nije jedna podskup druge, svaka ima svoje prednosti i mane)
- Kvantifikator “A” je implicitan. LTL formula je istinita za neki sustav (Kripke strukturu) ako vrijedi za sve moguce putove izvođenja.
- pomoću nje se opisuje kakav mora biti jedan linearni put. Takvi moraju biti svi mogući putevi da bi ona bila istinita.
- Linearna vremenska struktura je dana trojkom (S, x, L)
  - S: konačan skup stanja
  - x: N → S: beskonačna sekvenca stanja (za neki broj vraća sljedbenika)
  - L: S → 2^{AP}: pridruživanje simbola (labeling)
- minimalni skup: AND, NOT, U, X → ostalo se preko njih može izraziti
- LTL (za razliku od CTL) dozvoljava ugnježdivanje
- SEMANTIKA → pogledati slajd 9
- modaliteti beskonačnosti (nije moguće specificirati u CTL-u)
  - GF p (beskonačno često)
  - FG p (konačno globalno)
- distribucija preko logičkih vezica (to nije dozvoljeno u CTL)
- sa LTL ne možemo izraziti tvrdnje koje sadržavaju “moguće je”, “može” jer nema kvantifikator “E”

5.2. CTL*

- spaja CTL i LTL
- CTL i LTL su podskup od CTL*

5.3. Izračunavanje skupova stanja koji zadovoljavaju CTL formulu

- **eksplicitno** ili simbolički(BDD)
- P(V) tj. 2^V je skup svih podskupova od V (power set)
- u Kripke strukturi: relacija R je podskup od SxS
- slika (svi sljedbenici), pred-slika (svi prethodnici) → u Kripke strukturi
  - predslika od R je slika od R^{-1}
- mi imamo relaciju R koja za svako stanje definira sljedbenike. Nas će zanimati za svako stanje koji su njegovi sljedbenici i koji su njegovi prethodnici (slika i predslika) a to sve dobivamo naravno iz R.
- na slajdu 10 su nabrojane logičke rekurzije
- adekvatan skup: EX, EG, EU → sve se može na njih svesti
- Q(p) → skup svih stanja u kojima je p = TRUE
  - Q(TRUE) = S (sva stanja jer je u svim stanjima TRUE = TRUE)
  - Q(FALSE) = prazan skup (niti jedno stanje jer niti u jednom stanju nije FALSE = TRUE)
- **treba znati** izračunavanje skupova Q(EX f), Q(EG f), Q(E (f U G)) (**slajdovi 11, 12 i 13**)
  - za zadnja dva je potrebna teorija čvrste točke:
    - monotonost funkcije (definira se za funkciju F:P(S) → P(S)): ako X podskup od Y =>

F(X) podskup od F(Y)

- čvrsta točka je skup X za koji je F(X) = X. Samo će monotona funkcija imati fiksnu točku (i dvije(barem?): najveću i najmanju)
- Knaster – Tarski teorem (n+1 = |S|):
  - $F^i(Y)$ je monotona funkcija F primjenjena *i* puta na Y.
  - F^{n+1}(prazan skup) je najmanja čvrsta točka od F
  - $F^{n+1}(S)$ je najveća čvrsta točka od F
  - **dobivamo algoritam izračunavanja čvrste točke**:
    - najmanja čvrsta točka: primjenjujemo F na prazan skup sve dok ne nađemo čvrstu točku
    - najveća čvrsta točka: primjenjujemo F na S sve dok ne nađemo čvrstu točku
    - to se koristi tako da u rekurzivnim izrazima koje napišemo kao X = F(X) tražimo čvrstu točku od F → to je onda X po definiciji
    - **kad se računa EG onda koristimo najveću fiksnu točku, a kad se računa EU onda koristimo najmanju fiksnu točku**

- Izražavanje svega preko adekvatnog skupa (**EX, EG, EU**) :
  - EF p = **E** (True **U** p)
  - AG p = !(EF !p) = !(**E** (True **U** !p))
  - AF p = !(**EG** !p)
  - AX p = !(**EX** !p)
  - A (p U q) = !(**E** (!q **U** (!p ^ !q)) V **EG** !q)