Modelovanje sistema sa Kripke strukturama

Seminarski rad u okviru kursa
Verifikacija softvera
Matematički fakultet

Tatjana Damnjanović, 1046/2018
damnjanovic.tanja96@gmail.com

6. decembar 2018

Sažetak

Modelovanje sistema podrazumeva proces formiranja apstraktnog modela sistema na osnovu konkretne implementacije. Apstraktan model je graf čiji čvorovi predstavljaju stanja sistema, a grane prelaske između njih. Postoje tri vrste standardnih modela kojima se sistemi formalno opisuju. U ovom radu će biti opisan jedan od njih i data njegova primena.

Sadržaj

1 Uvod

Kako se informaciono-komunikacioni sistemi (eng. *Information and communications technology systems*) danas jako brzo razvijaju, postaju sve kompleksniji i široko su rasprostranjeni u svim sferama, neophodno je da budu pouzdani. Prilikom dizajniranja softvera i hardvera sistema više vremena se posvećuje verifikaciji, nego samoj konstrukciji. Formalne metode imaju veliki potencijal integrisanja verifikacije u rane faze procesa dizajniranja sistema, povećavaju efikasnost i smanjuju vreme verifikacije. Cilj formalnih metoda je da utvrde ispravnost sistema sa matematičkom preciznošću [1].

Verifikacione tehnike zasnovane na modelima opisuju sva moguća ponašanja sistema na matematički precizan i nedvosmislen način. Ispostavlja se da - pre bilo kakvog oblika verifikacije - precizno modeliranje sistema često dovodi do otkrivanja nepotpunosti, nejasnoća i nedoslednosti u neformalnim specifikacijama sistema. Kako je model sistema polazna tačka ovih tehnika, činjenica je da bilo koja verifikaciona tehnika koja se zasniva na modelima je dobra onoliko koliko i sam model sistema [1].

Proveravanje modela je verifikaciona tehnika koja ispituje sva stanja sistema primenom grube sile. Proveravanje modela zavisi od diskretnog modela kojim se sistem modeluje. Pošto graf nije dovoljno detaljan i ne pruža dovoljno informacija, koriste se neki drugi pristupi, od kojih se dva izdvajaju: Kripke strukture, kod kojih se čvorovi nazivaju izrazima, i labelirani tranzicioni sistemi, gde se grane nazivaju akcijama. Treba napomenuti da postoje i Kripke tranzicioni sistemi koji predstavljaju kombinaciju prethodna dva [4].

2 Tranzicioni sistem

Tranzicioni sistem se u računarstvu koristi kao model koji opisuje ponašanje sistema. Predstavlja se pomoću usmerenog grafa čiji čvorovi predstavljaju stanja sistema, a grane prelaske između tih stanja koje su labelirane.

Definicija 2.1. *Tranzicioni sistem je uređena šestorka* $(S, Act, \rightarrow, I, \nu, \lambda)$ *gde je:*

- S *skup stanja sistema,*
- Act *skup svih akcija u sistemu,*
- $\rightarrow \subseteq S \times Act \times S$ *relacija prelaska (u oznaci* $s_i \xrightarrow{\alpha} s_j$*, ukoliko akcija nije bitna onda se* α *može izostaviti),*
- $I \subseteq S$ *skup početnih stanja,*
- ν *skup iskaznih promenljivih (skup predikata),*
- $\lambda : S \rightarrow \mathbb{P}\nu$ *funkcija mapiranja (svakom stanju pridružuje skup predikata koji važe u tom stanju) [6].*

Napomena 2.1. Za tranzicione sisteme važi sledeće:

- Skup stanja može biti konačan ili beskonačan .
- Bez gubitka na opštosti može se pretpostaviti da je relacija $\rightarrow$ totalna, tj. važi:

$$(\forall s \in S)(\exists s' \in S)(s \rightarrow s')$$

- Sistem može biti deterministički (po akcijama) ako važi:

$$(\forall s \in S)(\forall \alpha \in Act)(\exists! s' \in S)(s \xrightarrow{\alpha} s')$$

  i nedeterministički.
- Kod determinističkih se podrazumeva tačno jedno početno stanje [6].

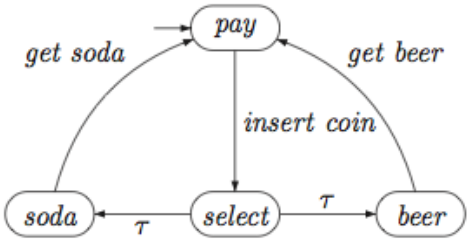
Definicija 2.2. *Putanja u sistemu* $T = \{S, Act, \rightarrow, I, \nu, \lambda\}$ *je beskonačni niz koji alternira stanja i akcije* $\sigma = s_0\alpha_1 s_1 \alpha_2 s_2 ...$ *takav da važi* $s_i \xrightarrow{\alpha_{i+1}} s_{i+1}$ *za svako* $i \geq 0$.
Izvršavanje u sistemu T *je bilo koja putanja* σ *čije je početno stanje* $s_0 \in S$.
Stanje S *je dostižno ako postoji izvršavanje* σ *takvo da je* $s \in \sigma$ *[6].*

Primer 2.1. Na slici 1 je dat primer tranzicionog sistema iz [1] gde je:

- $S = \{pay, soda, select, beer\}$ skup svih stanja sistema,
- $Act = \{insert\ coin, get\ beer, get\ soda, \tau\}$ skup akcija u sistemu,
- $I = \{pay\}$ inicijalno stanje sistema,
- Neke tranzicije su: $pay \xrightarrow{insert\ coin} select$, $beer \xrightarrow{get\ beer} pay$, $soda \xrightarrow{get\ soda} pay$,
- ν zavisi od osobine koja se razmatra. Na primer, ako se razmatra osobina: „Mašina daje piće samo nakon što dobije novčić", onda je $\nu = \{paid, drink\}$,
- $\lambda(pay) = \emptyset$, $\lambda(select) = \{paid\}$, $\lambda(beer) = \lambda(soda) = \{paid, drink\}$.

Primeri izvršavanja u sistemu T su:

- $\rho_1 = pay \xrightarrow{insert\ coin} select \xrightarrow{\tau} soda \xrightarrow{get\ soda} pay \xrightarrow{insert\ coin} select \xrightarrow{\tau} beer \xrightarrow{get\ beer} pay...$,
- $\rho_2 = pay \xrightarrow{insert\ coin} select \xrightarrow{\tau} beer \xrightarrow{get\ beer} pay \xrightarrow{insert\ coin} select \xrightarrow{\tau} beer \xrightarrow{get\ beer} pay...$,
- $\rho_3 = select \xrightarrow{\tau} soda \xrightarrow{get\ soda} pay \xrightarrow{insert\ coin} select \xrightarrow{\tau} beer \xrightarrow{get\ beer} pay...$



Slika 1: Primer tranzicionog sistema

3 Kripke struktura

Kripke struktura je vrsta tranzicionog sistema koji je originalno razvio Saul Kripke, a koja se koristi u proveravanju modela za predstavljanje ponašanja sistema. Neformalno, to je graf čiji su čvorovi dostupna stanja sistema, a ivice predstavljaju prelaske između stanja sistema. Funkcija labeliranja (mapiranja) je funkcija koja preslikava svaki čvor u skup osobina koje važe u odgovarajućem stanju [7].

U nastavku sledi formalna definicija i primeri modelovanja sistema sa Kripke strukturama.

3.1 Definicija

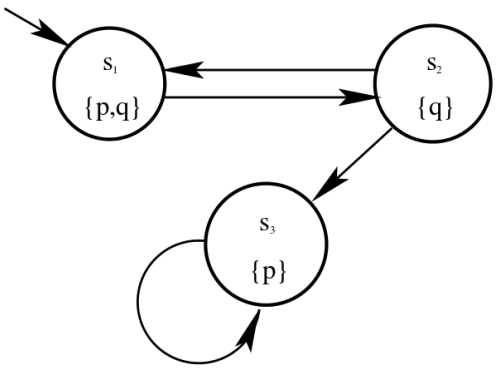
Definicija 3.1. *Kripke struktura predstavlja uređenu četvorku* $M =< S, S_0, R, L >$ *gde je:*

- S *konačan skup stanja,*
- $S_0 \subseteq S$ *skup početnih stanja,*
- $R \subseteq S \times S$ *relacija prelaska, koja mora biti totalna,*
- $L : S \to 2^{AP}$ *funkcija (labeliranja, mapiranja) koja preslika svako stanje u skup atomičnih formula (eng.* atomic propositions*) AP koje su tačne u tom stanju [5].*

Definicija 3.2. *Putanja u Kripke strukturi M predstavlja beskonačan niz stanja* $\pi = s_0 s_1 ... s_n ...$ *takvih da* $(\forall i > 0)\ (s_i, s_{i+1}) \in R$ *[5].*

3.2 Primeri

Primer 3.1. Neka je skup atomičnih formula $AP = \{p, q\}$ gde p i q mogu predstavljati proizvoljno binarno svojstvo sistema koji se modelira Kripke strukturom.



Slika 2: Primer Kripke strukture

Na slici 2 je prikazana Kripke struktura $M =< S, I, R, L >$ gde je:

- $S = \{s_1, s_2, s_3\}$ skup svih stanja,
- $I = \{s_1\}$ inicijalno stanje,

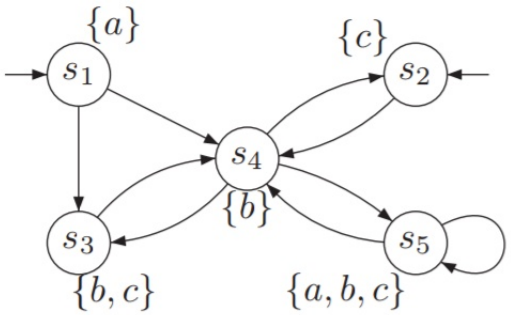
- $R = \{(s_2, s_2), (s_2, s_1), (s_2, s_3), (s_3, s_3)\}$ relacija prelaska,
- $L = \{(s_1, \{p, q\}), (s_2, \{q\}), (s_3, \{p\})\}$ funkcija labeliranja.

Jedna od mogućih putanja je $\rho = s_1, s_2, s_1, s_2, s_1, s_2, s_3, s_3, s_3, ...$, a reč koju ova putanja proizvodi je $w = \{p, q\}, \{q\}, \{p, q\}, \{q\}, \{p, q\}, \{q\}, \{p\}, \{p\}, \{p\}, ...$ [7].

Primer 3.2. Neka je $AP = \{a, b, c\}$ skup atomičnih formula, gde a, b, c predstavljaju neka binarna svojstva sistema.

Na slici 3 je ilustrovana Kripke struktura $M =< S, I, R, L >$ gde je:

- $S = \{s_1, s_2, s_3, s_4, s_5\}$ skup svih stanja,
- $I = \{s_1, s_2\}$ inicijalna stanja,
- $R = \{(s_1, s_3), (s_1, s_4), (s_2, s_4), (s_3, s_4), (s_4, s_2), (s_4, s_3), (s_4, s_5), (s_5, s_4), (s_5, s_5)\}$ relacija prelaska,
- $L = \{(s_1, \{a\}), (s_2, \{c\}), (s_3, \{b, c\}), (s_4, \{b\}), (s_5, \{a, b, c\})\}$ funkcija labeliranja.



Slika 3: Primer Kripke strukture [1]

Kripke strukture su usko povezane sa automatima sa konačnim brojem stanja. Ukoliko se uzme da Kripke struktura ima samo jedno inicijalno stanje, onda se može poistovetiti sa Murovom mašinom [3].

Napomena 3.1. Kripke struktura određuje samo stanje i izračunavanje sistema, ne i način na koji se do toga došlo. To znači da Kripke struktura ne objašnjava zašto je sitem u određenom stanju ili zašto se pomera u neko drugo stanje. Kripke strukture ne prave razliku između ulaza, izlaza, lokalnih promenljivih i programskih lokacija. Umesto toga one prikupljaju sve moguće vrednosti različitih promenljivih koje se mogu pojaviti prilikom izračunavanja sistema [3].

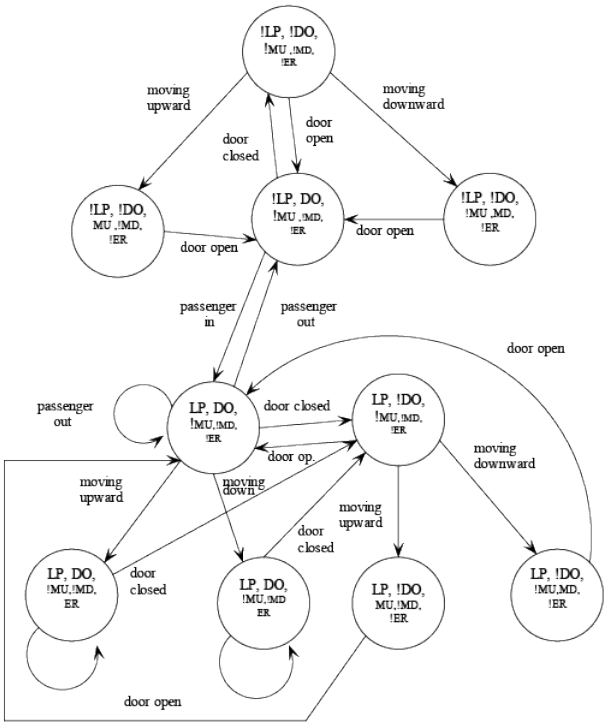
4 Primer modelovanja sistem lifta

Ponašanje sistema lifta se može modelovati kao na slici 4. Da bi ilustracija bila jasnija, u svakom stanju je obeležen svaki iskaz koji važi, kao i

negacija svih onih koji ne važe u tom stanju. Prelazi su labelirani da bi se prikazale akcije kojima se prelazi u drugo stanje, a inače to ne predstavlja deo Kripke strukture. Iskazi koji opisuju stanje sistema lifta su sledeći:

- MU : lift se kreće na gore,
- MD : lift se kreće na dole,
- DO : vrata lifta su otvorena,
- LP : putnici su ušli u lift,
- ER : greška.

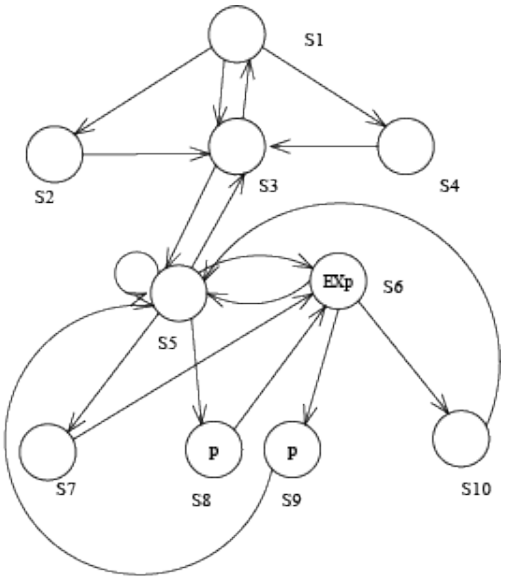
Na slici 5 je prikazana pojednostavljena Kripke struktura sistema lifta koja se dobija iz prethodne ilustracije Kripke strukture sa labeliranim prelazima (slika 4) imenovanjem stanja i uklanjanjem labela sa prelaza. Takođe, zbog jednostavnosti, uklonjeni su neki prelazi koji suštinski nisu bitni za sam sistem [2].



Slika 4: Kripke struktura sa labeliranim prelazima

Elementi Kripke strukture koja modeluje ponašanje sistema lifta su:

- $S = \{s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}\}$ skup svih stanja,
- $I = \{s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}\}$ skup inicijalnih stanja,
- $R = \{(s_1, s_2), (s_1, s_3), (s_1, s_4), (s_2, s_3), (s_3, s_1), (s_3, s_5), (s_4, s_3), (s_5, s_5), (s_5, s_3), (s_5, s_7), (s_5, s_8), (s_5, s_6), (s_6, s_5), (s_6, s_9), (s_6, s_{10}), (s_7, s_6), (s_8, s_6), (s_9, s_5), (s_{10}, s_5)\}$ relacija prelaska,



Slika 5: Pojednostavljena Kripke struktura sistema lifta

- $L = \{(s_1, \{!LP, !DO, !MU, !MD, !ER\}),$
  $(s_2, \{!LP, !DO, MU, !MD, !ER\}), (s_3, \{!LP, DO, !MU, !MD, !ER\}),$
  $(s_4, \{!LP, !DO, !MU, MD, !ER\}), (s_5, \{LP, DO, !MU, !MD, !ER\}),$
  $(s_6, \{LP, !DO, !MU, !MD, !ER\}), (s_7, \{LP, DO, !MU, !MD, ER\}),$
  $(s_8, \{LP, DO, !MU, !MD, ER\}), (s_9, \{LP, !DO, MU, !MD, !ER\}),$
  $(s_1, \{LP, !DO, !MU, MD, !ER\})\}$ funkcija labeliranja.

5 Zaključak

U radu je opisan tranzicioni sistem kao jedan od modela kojim se modeluje ponašanje sistema, a zatim i Kripke struktura kao vrsta tranzicionog sistema. Navedene su njihove formalne definicije i četiri manja primera, kao i jedan veći primer koji se bavi modelovanjem ponašasanja sistema lifta. Napomenuto je da Kripke struktura ne određuje način na koji se došlo do određenog stanja, već samo stanje i trenutni rezultat sistema. Dato je poređenje Kripke strukture sa konačnim automatima. Problem koji se može javiti kod modelovanja sistema sa Kripke strukturom je problem validacije (eng. *validation problem*). To znači da, i pored toga što su prethodno opisani koraci kreiranja modela jasni, u praksi može nastati problem ocenjivanja da li je model sa svojim svojstvima adekvatan opis ponašanja sistema. Ovaj problem se javlja kada su u pitanju kompleksni sistemi ili su sistemske funkcionalnosti neprecizno i neformalno opisane. Da bi se ovaj problem umanjio ili u potpunosti prevazišao, potrebno je svojstva modela opisati precizno i nedvosmisleno. To se radi korišćenjem jezika za opisivanje svojstava (eng. *property specification language*). Da bi se poboljšao kvalitet modela, bilo bi dobro da se pokrene simulacija pre samog proveravanja modela. Simulacijom se mogu pronaći i eliminisati jednostavnije greške koje se prave prilikom modelovanja. Eliminisanjem

ovih jednostavnih grešaka pre bilo kakvog oblika temeljne provere može se smanjiti skupa i dugotrajna verifikacija [1].

Literatura

[1] C. Baier, J.P. Katoen, and K.G. Larsen. *Principles of Model Checking*. Mit Press. MIT Press, 2008.

[2] Dr. Jatindro Kr. Deka and Dr. Santosh Biswas. Module VI: Model Checking. on-line at: https://nptel.ac.in/courses/106103016/module6/lec2/5.html.

[3] Klaus Schneider. *Verification Of Reactive Systems - Formal Methods and Algorithms*. Springer-Verlag, 2004.

[4] B. Steffen, D. Schmidt, and M. Muller-Olm. Model Checking - A Tutorial Introduction, 1999. on-line at: https://www.researchgate.net/publication/221477319_Model-Checking_A_Tutorial_Introduction.

[5] E. Torlak. Model Checking I - slides, 2014. on-line at: https://courses.cs.washington.edu/courses/cse507/14au/slides/L14.pdf.

[6] M. Vujošević Janičić. Verifikacija softvera - Proveravanje modela. on-line at: http://www.programskijezici.matf.bg.ac.rs/vs/predavanja/07_proveravanje_modela/proveravanje_modela_slajdovi.pdf.

[7] Wikipedia. Kripke structure (model checking). on-line at: https://en.wikipedia.org/wiki/Kripke_structure_(model_checking).