

DODATNI PRIMERJALNI TESTI ZA SIMULATORJE SPICE

Tomaž Dogša

Fakulteta za elektrotehniko, računalništvo in informatiko, Univerza v Mariboru

Ključne besede: SPICE orodja računalniška, testi primerjalni, simulatorji računalniški, točke delovne, DC analize enosmerne, testiranja avtomatska, avtomatizacija testiranja, vezja naključna

Povzetek: Dosedanji primerjalni testi, s katerimi lahko primerjamo kakovost simulatorjev, temeljijo na posebej izbrani majhni množici vezij. V prispevku predlagamo dodatne skupine primerjalnih statističnih testov in kriterije, s katerimi bi lahko na objektivni način primerjali kakovost simulatorjev električnih vezij. Vsa vezja, ki so vključena v dodatno množico testov imajo naključno tvorjeno strukturo in naključno izbrane elemente in parametre. Ker lahko s posebnim generatorjem vezij tvorimo poljubno število takih vezij, je možno podajati statistično ovrednotena kvantitativna merila glede kakovosti simulatorjev.

Additional Benchmark Tests for SPICE Simulators

Keywords: SPICE computer tools, Simulation Program with Integrated Circuit Emphasis computer tools, benchmarks, comparative tests, computer simulators, biases, operating points, DC analyses, Direct Current analyses, automatic testing, test automation, random circuits

Abstract: The purpose of benchmark test is to compare the quality attributes of circuits simulators. Existing benchmark tests are based on selected small set of electronic circuits. In this paper we propose additional sets of statistically based benchmark tests and corresponding criteria that enable unbiased comparison. All test circuits that form additional benchmark sets have randomly generated structure, randomly chosen elements and their parameters. Statistically quantization of quality attributes is now possible, since high number of circuits can be generated by the special software tool.

1. Uvod

Načrtovanje elektronskih vezij brez pomoči računalnika je danes praktično neizvedljivo. Vse večja kompleksnost vezij in nelinearnost elektronskih elementov v večini primerov onemogočajo analitični pristop. Načrtovalci si v takih primerih pomagajo s simulatorji elektronskih vezij. Simulatorji so zelo kompleksni programski paketi, ki se neprestano spreminjajo: izboljšujejo se uporabljeni algoritmi in modeli ter dodaja se nova funkcionalnost. Eden izmed glavnih namenov simulatorja je napovedovanje obnašanja realnega vezja. Kljub relativni zrelosti simulatorjev, se dogaja, da dajejo napačne rezultate (glej npr. /PERŠIČ,1995/). Na tržišču obstaja več vrst simulatorjev, ki se med seboj razlikujejo po funkcionalnosti, uporabljenih algoritmi in modelih, ceni in drugih atributih kakovosti. Njihova medsebojna primerjava temelji na določenih primerjalnih testih (benchmark). Kaj je primerjalni test? Davidson in Harlow v svojem uvodniku /DAVIDSON,2000/ ponujata delno prirejeno definicijo, ki jo lahko najdemo v leksikonu Merriam-Webster Collegiate Dictionary: "Primerjalni test je standardiziran problem (vezje ali del vezja), ki ga uporabljamo za primerjavo zmogljivosti (hitrost, učinkovitost, kakovost rezultata) različnih orodij in algoritmov".

Namen primerjalnih testov ni samo primerjanje izbranih karakteristik (npr. hitrosti) ampak širši vpogled v kakovost simulatorja. Zaradi tega so tudi zanimivi za načrtovalce in preverjevalce EDA (Electronic Design Automation) programske opreme. S primerjalnimi testi lahko testirajo simulatorje ali pa jih uporabijo za merjenje uspešnosti novih ali izboljšanih algoritmov. Vsak primerjalni test mora biti sestavljen iz opisa postopka, opisa testnih primerov in kriterijev za vrednotenje primerjave. Testne primere tvorijo vhodni

podatki in pričakovani rezultati oziroma pričakovano obnašanje.

Primerjalni testi naj bi predstavljali reprezentativne primerke iz posameznih problemskih razredov. Npr. zelo znana zbirka 58 vezij, ki so jo leta 1990 predlagali na eni izmed delavnic na MCNC (Microelectronic Center of North Carolina. (<http://www.cbl.ncsu.edu/>), je sestavljena in petih skupin. Te skupine lahko glede kompleksnosti vezij razdelimo v dve grupi: skupina majhnih (približno do 500 elementov) in skupina obširnejših (nad približno 1000 elementov) vezij. Vezja so tako izbrana, da pokrivajo celotno problemsko domeno.

Slabost sedanjih primerjalnih testov oziroma metod kritizira tudi F. Brglez /BRGLEZ,2000/. Zavzema se za znanstveni pristop na področju eksperimentiranja, kar pomeni, da se ne oziramo na nekaj izbranih testnih vezij ampak na množico. S statistično analizo nato ovrednotimo rezultate eksperimenta oziroma primerjalne analize. Zato predlagamo, da se obstoječemu nizu primerjalnih testov dodajo še tri skupine. Tako bi za potrebe primerjalne analize in testiranja uporabljali naslednje skupine:

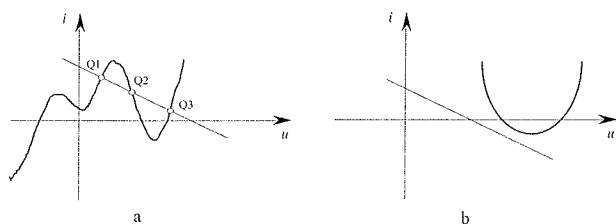
1. skupina: izbor posameznih konkretnih vezij (obstoječa skupina),
2. skupina: množica naključno generiranih topološko pravih vezij,
3. skupina: izbor posameznih topološko nepravilnih vezij,
4. skupina: množica naključno generiranih topološko nepravilnih vezij.

Vključitev skupine s topološko nepravilnimi vezji je smiselna zaradi tega, ker pri takih vezjih rešitev (niti teoretična) ne more obstajati in nas mora o tem simulator obvestiti. V pris-

pevku bomo najprej opisali določeno problematiko, ki je povezana z naključnim generiranjem vezij. Kljub temu, da se bomo omejili, le na simulatorje analognih vezij to še ne pomeni, da predlaganih idej ni možno posplošiti na digitalna in analogno-digitalna vezja. Vsak simulator nudi uporabniku več vrst analiz. Najbolj osnovna je vsekakor analiza delovne točke. Ker predstavlja izhodišče za skoraj vse druge analize, smo se osredotočili samo na njo. Torej, zanimali nas bodo primerjalni testi, s katerimi bi lahko ovrednotili uspešnost algoritmov za analizo delovne točke. Najprej bomo na kratko opisali probleme, ki se utegnejo pojaviti pri analizi delovne točke nato pa problematiko generiranja naključnih vezij. Za ilustracijo bomo prikazali tudi konkretne rezultate primerjalne analize za tri simulatorje: SPICE2G6 (verzija Intusoft IS SPICE 1.41 12/12/87, PSPICE (verzija 3.01, januar 1987) in SPICE3F4 (verzija 18. februar 1999). Izbor je bil povezan predvsem z njihovo dosegljivostjo. Če bi uporabili najnovejše verzije, bi zagotovo dobili drugačne rezultate.

2. Delovna točka

Delovna točka je niz takšnih parov enosmernih tokov in napetosti, ki hkrati zadoščajo karakteristiki elementov in obema Kirchhoffovima zakonoma. Ker gre za enosmerno analizo, simulator najprej pretvori vezje v rezistivno variantno in nato izvede simulacijo. Vsi algoritmi, ki jih uporabljajo simulatorji analognih vezij za analizo delovne točke, temeljijo na določenih iterativnih metodah, ki jih omejimo s številom iteracij in zahtevano natančnostjo rezultata. Pri analitičnem ali grafičnem izračunu vidimo, da lahko obstaja ena, več ali pa nobena rešitev (slika 1 a in b).



Slika 1 Zgled grafične analize dveh vezij, ki sta sestavljeni iz baterije, upora in nelinearnega elementa. Vezje (a) tri teoretične rešitve, vezje (b) nima teoretične rešitve

Več rešitev več ne moremo interpretirati kot rezultat enosmerne analize, ampak kot stacionarno stanje prehodnega pojava električnega vezja. Rešitev torej ni odvisna samo od vezja in nastavitve parametrov, s katerimi vplivamo na algoritem, ampak tudi od začetnih pogojev. Ena izmed slabosti tipičnih simulatorjev analognih vezij (npr. SPICE) je ta, da izračunajo samo eno delovno točko, kljub temu, da obstaja več rešitev. Obstajajo tudi simulatorji, ki z uporabo popolnoma drugačnih algoritmov, izračunajo vse delovne točke oziroma ravnotežna stanja. Najbolj znani metodi sta: iterativna odsekoma-linearna kombinacijska analiza /CHUA, 1975/ in homotopna metoda /TRAJKOVIC, 1991/. Sled-

nja je tudi implementirana v simulatorju HomSPICE /TRAJKOVIC, 1998/, ki pa ga nismo uspeli preizkusiti.

Eden izmed pogojev, da bo simulator poiskal delovno točko, je obstoj teoretične rešitve. Neobstoj teoretične rešitve se lahko pojavlja samo pri vezjih, ki vsebujejo enega ali več nerealno modeliranih elementov. Glede oblike enosmernih karakteristik modelov ni nobenih omejitev razen za vrednosti toka oziroma napetosti, ko le-ta limitira proti neskončnosti. Ker vedno obstaja neka delovna točka, pri kateri postane vezje totalno pasivno, pomeni, da se morajo karakteristike dvopolnih elementov začeti v prvem in končati v tretjem kvadrantu. Takoj vidimo, da idealni napetostni in tokovni vir te zahteve ne izpolnjujeta. Ta problem lahko simulator reši z dodajanjem ustreznega upora k vsakemu napetostnemu ali/in tokovnemu viru in z dodatnimi zahtevami glede topologije (npr. zanka iz samih napetostnih virov je prepovedana). Če karakteristika vsakega elementa zadosti prej omenjenemu kriteriju, potem vedno obstaja teoretična rešitev. Če vsaj en element tega kriterija ne izpolnjuje, se lahko zgodi, da teoretična rešitev ne obstaja (slika 1 b).

Vsa pravkar opisana problematika se uporabniku kaže v neuspešnosti analize, ki se večinoma navzven pokaže kot nekonvergirano rešitev. Problem konvergence je možno reševati na več načinov:

1. Simulatorju pomagamo s podatkom o približni rešitvi (stavek .NODESET).
2. Spremenimo modele - uporabimo realnejše modele.
3. Spreminjamo razne parametre, s katerimi vplivamo na potek reševanja (npr. povečamo največje število iteracij).
4. Uporabimo počasni naraščajoče napetostne vire (source-stepping algorithms).
5. V vsako vozlišče dodamo parazitno prevodnost, ki jo počasi manjšamo (Gmin-stepping).
6. Z dodanimi parazitnimi reaktivnimi elementi pretvorimo vezje v dinamično in simuliramo vklop vezja (pseudo-transient analysis).
7. Izberemo drugačen algoritem za analizo delovne točke (uporabimo npr. odsekoma-linearno kombinacijsko analizo ali homotopno metodo).

Pravilnost izračuna delovne točke je odvisna od pravilnosti uporabljenih modelov in pravilnosti ter učinkovitosti algoritma. Pri sodobnih simulatorjih je lahko vzrok za napačen izračun delovne točke tudi v napačnem delovanju postprocesorja. V tem primeru jedro simulatorja izračuna pravilno vrednost, grafični postprocesor jo pa izpiše napačno.

3. Odpoved simulatorja pri izračunu delovne točke

Izračun delovne točke je ena izmed temeljnih analiz, ki jih izvaja simulator. Nepravilnosti pri izračunu delovne točke vplivajo na pravilnost večine analiz (npr. .AC, .TF, .DC, .TRAN). Potem, ko smo sprožili zahtevo za analizo delovne točke, lahko simulator preide v naslednja stanja:

1. Delovna točka je izračunana in **se nahaja** znotraj dovoljenih odstopanj pričakovane vrednosti.
2. Delovna točka je izračunana vendar se **ne nahaja** znotraj dovoljenih odstopanj pričakovane vrednosti. V tem primeru gre za alternativno delovno točko oziroma za vezje, ki ima več delovnih točk. Možno je tudi, da simulator deluje nepravilno.
3. Delovna točka **ni izračunana**, kljub temu da ima vezje pravilno topološko strukturo (npr. rešitev ne konvergira ali pa nastopi popolna odpoved simulatorja)
4. Delovna točka ni izračunana, ker ima vezje nepravilno topološko strukturo (npr. zanka iz samih napetostnih virov).

Stanje številka 3 bomo klasificirali kot odpoved. Popolna odpoved simulatorja se pojavi takrat, ko delovanje programa nasilno ustavi operacijski sistem. Za firmo, ki je izdelala simulator je to sicer zelo neprijetna odpoved, za uporabnika pa zgolj moteča in hkrati nenevarna, saj je le-ta z odpovedjo seznanjen. Kritične oziroma fatalne so lahko tiste odpovedi, katerih uporabnik ne opazi. To se lahko zgodi, če ne razpolaga s pričakovano vrednostjo oziroma je interval, v katerem pričakuje rezultat, prevelik.

4. Naključno analogno vezje

Naključna vezja lahko generiramo na dva načina. Postavimo nek splošen model, ki ustreza neki skupini vezij in nato s spreminjanjem parametrov in nekaterih delov strukture tvorimo množico naključnih vezij. Zgled za to metodo bi lahko bil npr. splošen model enostopenjskega diferenčnega ojačevalnika, ki vključuje tudi nekaj variant. Pri drugi metodi naključno generiramo graf vezja in nato elemente. S to metodo lahko v bistvu nastane katerokoli možno vezje, pri prvi pa smo omejeni z izbranim splošnim modelom.

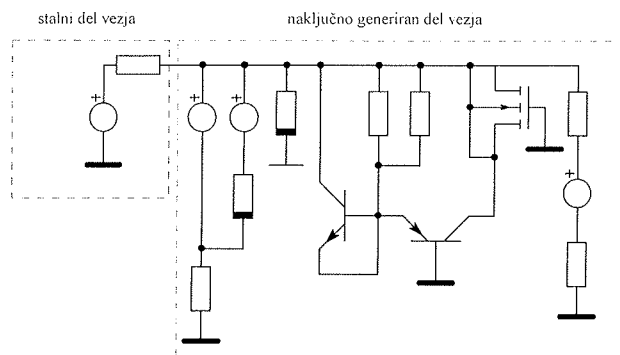
Za generiranje naključnih vezij smo izdelali poseben program (generator naključnih vezij), ki delno združuje oba pristopa. Naključno vezje, ki ga tvori generator, je sestavljeno iz stalnega in naključnega dela. Najpomembnejši podatki, ki vplivajo na strukturo naključnih vezij in vrednosti parametrov, so:

1. Deleži posameznih elementov in njihovi parametri (R, L, C, E, G, I, V, D, NMOS, PMOS, QN, QP). E in G sta polinomska vira, ki ju uporabljamo za modeliranje krmiljenih virov oziroma nelinearnih dvopolov. Parametri so: največja in najmanjša vrednost ter vrsta porazdelitve. Trenutna verzija ne omogoča naključni izbor parametrov v .MODEL stavkih.
2. Kompleksnost vezja (število vozlišč, število elementov).
3. Odločitev o generiranju topološko pravih ali nepravilnih strukturah.
4. Izbor porazdelitvene funkcije po kateri se vrši izbor naključnih entitet (Gaussova, konstantna ali kaotična porazdelitev).
5. Statistična privilegiranost dveh vozlišč. Pri večini realnih vezij lahko vidimo, da incidenčna matrika ni ena-

komerno napolnjena oziroma nekatera vozlišča izrazito odstopajo glede števila priključenih elementov. To sta vozlišči, kamor je priključena napajalna napetost in masa (glej sliko 3).

Na videz je funkcija generatorja podobna tisti, ki se uporablja pri Monte Carlo analizi. Razlika je v tem, da tukaj naključno spreminjamo strukturo vezja, elemente in parametre, medtem ko pri Monte Carlo analizi samo vrednosti parametrov.

Glede na izbor podatkov, ki so vhod v generator, lahko generiramo zelo pestro množico vezij. Z nelinearnimi upori, ki so implementirani z G viri (za SPICE3 se uporablja B element), generator simulira uporabo zunanjih vedenjskih modelov. Z izborom lihe oziroma sode stopnje polinoma generiramo teoretično nemogoče (npr. element na sliki 1 b) oziroma mogoče nelinearne upore (npr. element na sliki 1 a). S temi upori lahko tudi spreminjamo nelinearnost vezja. Zgled naključno generiranega vezja prikazujeta sliki 2 in 3.



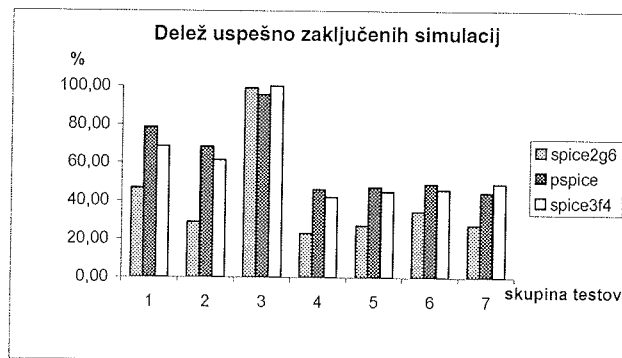
Slika 3 Shema naključno generiranega analognega vezja narisane na podlagi opisa iz slike 2. Stalni del vezja je enak za vsa generirana vezja.

5. Rezultati primerjalnega testiranja

Vsako testiranje izvajamo po naslednjih korakih: tvorjenje testnih vzorcev, tvorjenje pričakovih vrednosti, izvedba analize in vrednotenje rezultatov. Vse korake razen drugega je relativno enostavno avtomatizirati. Za avtomatizacijo drugega koraka moramo napraviti zahteven program za sintezo vezja. V primeru testiranja več simulatorjev imamo tudi možnost, da enega izberemo za referenčnega. Glede na predlagane skupine vezij namenjenih primerjalnemu testiranju, smo izvedli samo testiranje z množico naključno generiranih topološko pravih vezij.

Tabela 1 Opis testnih vezij

Testna garnitura	število vezij	opis
1	169	vsi elementi, brez nelinearnih uporov
2	200	samo NMOS, PMOS, D, R in napajalni viri
3	199	samo QN, QP, D, R in napajalni viri
4	169	vsi elementi, nelinearnost uporov do 4.stop.
5	169	vsi elementi, nelinearnost uporov do 3.stop.
6	168	vsi elementi, nelinearnost uporov do 2.stop.
7	106	brez polprevodnih elementov, nelinearnost uporov do 4.stop



Slika 4 Delež uspešno zaključenih simulacij

Ker bi opis in diskusija rezultatov za vse primerjalne teste presegala dovoljen obseg prispevka, se bomo osredotočili le na sedmo garnituro testov, v kateri ni nobenih polprevodnih elementov. S tem smo izločili morebiten vpliv kakovosti

```

VEZ117.cir
*SPICE_NET
*Stevilka vezja:
*117
*Datum nastanka: 4.4.2001 Generator vezij:V5.0 2.4.2001
*Konfiguracijska datoteka: k4g2k1.dat
*Nastanek konfiguracijske datoteke: 4.4.2001 12:32:24
*Verzija opisovanja nel. elementov:2 Stevilo vozlišc: 6 Stevilo el.: 14
*Nacin generiranja vezja: naključno
*Statistično poudarjeno vozlišce: 0 20.00%
*Statistično poudarjeno vozlišce: 1 20.00%

R1 0 5 1.04222864689678E+0004
R2 1 3 1.64998529084921E+0004
R3 2 0 7.76117287476063E+0004
V4 3 2 -9.63017179630697E+0000
R5 4 1 5.58461088487506E+0003
E6 5 1 POLY(1) (4 6) 3.88584041036665E-0001 -4.95419465005398E-0002
+ -1.17212752811611E-0001 -5.41528668254614E-0002
+ -6.52682286687195E-0001
G7 6 5 POLY(1) (6 5) -1.81034177541733E-0001
+ -4.58973378874362E-0001 + 2.83149098977447E-0001
QN8 4 0 1 QN
R9 1 4 3.21886612679362E+0004
E10 6 1 POLY(1) (2 1) -3.01108809188008E-0001
+ -5.59775688685477E-0001 + 4.59779966622591E-0001
QP11 1 3 3 QP
G12 0 1 POLY(1) (0 1) -6.98641702532768E-0001 5.45442000962794E-0001
+ 8.75892678275704E-0001 -1.84644504450262E-0001 -7.96733874827623E-0001
MP13 1 0 1 1 PENH L=8U W=7U
QN14 1 2 2 QN
.OP
* ===== Dodana datoteka : MOD_VSI.MOD =====
* Posebej dodamo nap. vir zato, da bo vedno prisoten vsaj en
VDD 5000 0 5
RG 5000 1 1
.OPTIONS NOMOD

*****
** ( MODEL M8CN-290) **
** MCNC 0.8 CMOS PROCESS PISCES **
** ( NOMINAL ) FEBRUARY 1990 **
** SOURCE: CIRCUITSIM90 BENCHMARK CIRCUITS FILE MODELS.NOM **
*****

```

Slika 2 Zgled naključno generiranega analognega vezja

notranjih modelov in se osredotočili predvsem na učinkovitost algoritma za izračun delovne točke. Statistična porazdelitev elementov je bila naslednja: R (40%), V (10%), I (10%), G (20%), E (20%). Z napetostno krmiljenimi tokovnimi viri (G) smo realizirali nelinearne upore, katerih naključno generirana karakteristika je definirana s polinomom. Največja stopnja polinoma je bila 4. Podobno velja za krmiljen nelinearni napetostni vir E. Za vrednotenje primerjalnega testa smo postavili tri merila: merilo uspešnosti, merilo podobnosti in merilo ekskluzivnosti.

Merilo uspešnosti je delež uspešno zaključenih simulacij. Za učinkovito primerjalno testiranje morajo biti testna vezja tako izbrana, da je uspešnost vedno manjša od 100%, sicer nastopi pojav nasičenja (merilo več ne zaznava sprememb). Če se uspešnost začne približevati 100%, je potrebno povečati število vezij oziroma stopnjo nelinearnosti.

Zanimiva sta tudi atributa *skupna napaka* in število *alternativnih rešitev*. Razliko med referenčno in dejansko vrednostjo delovne točke vrednotimo s povprečnim kvadratičnim pogreškom. Če je bil ta večji od nekega izbranega praga (5V), potem smo izračunano delovno točko vezja kvalificirali kot alternativno. Skupna napaka je vsota vseh napak vezij, ki so bila uspešno simulirana (izvzeta so bile alternativne delovne točke). To merilo je seveda problematično, saj je za naključno vezje težko podati enoten kriterij, ki bi omogočal identifikacijo alternativnih točk.

Tabela 2 Podrobni rezultati za sedmo garnituro testov.

	SPICE2G6	PSPICE	SPICE3F4
število neuspešnih simulacij	73%	56%	51%
število uspešnih simulacij	27%	44%	49%
skupna napaka	27V	31V	-
štev. altern. rešitev	2%	9%	-

Merilo podobnosti: s to metriko vrednotimo podobnost v obnašanju simulatorjev. Izračunamo ga kot kvocient med številom enakih stanj, v katero prideta dva simulatorja in številom testov. Če je podobnost med dvema simulatorjema enaka 1, pomeni, da sta bila uspešna in neuspešna pri enakih testnih primerih (vezjih). Pri testni garnituri številka 7 sta se SPICE3F4 in PSPICE v 87% testov obnašala enako (tabela 3).

Tabela 3 Merilo podobnosti za sedmo garnituro testov.

	spice2g6	pspice	spice3f4
spice2g6	1	0,83	0,76
pspice	0,83	1	0,87
spice3f4	0,76	0,87	1

Merilo ekskluzivnosti: To merilo se nanaša na uspešnost tistih vezij, pri katerih je določen simulator odpovedal. Npr.

SPICE2G6 je odpovedal pri 77 vezjih (tabela 4) in v množici teh vezij je PSPICE uspešno simuliral 23% vezij, SPICE3F4 pa 30%. Hkrati tudi vidimo, da vsa vezja, ki so povzročila odpoved simulatorja SPICE3F4, so tudi povzročila odpovedi ostalih simulatorjev. Ali z drugimi besedami, tam kjer SPICE3F4 ni uspel, nista uspela tudi ostala simulatorja.

Z drugimi skupinami testov smo ugotovili, da to vedno ne velja. Npr. pri testni garnituri štev. 2 (tabela 5), je PSPICE uspel poiskati rešitev za 37% vezij, za katere je bil SPICE3F4 neuspešen. Kljub temu lahko glede na vrednost ekskluzivnosti, še vedno lahko sklepamo, da je SPICE3F4 najboljši.

Tabela 4 Merilo ekskluzivnosti za sedmo garnituro testov.

	spice2g6	pspice	spice3f4
spice2g6	77 (100%)	18 (23%)	23 (30%)
pspice	0 (0%)	59 (100%)	5 (9%)
spice3f4	0 (0%)	0 (0%)	54 (100%)

Tabela 5 Merilo ekskluzivnosti za drugo garnituro testov.

	spice2g6	pspice	spice3f4
spice2g6	142 (100%)	84 (59%)	66 (46%)
pspice	5 (8%)	63 (100%)	15 (24%)
spice3f4	1 (1%)	29 (37%)	77 (100%)

7. Sklep

Tradicionalno primerjalno testiranje temelji na izbrani majhni množici tipičnih vezij. Ker vezja niso izbrana naključno in ker jih je premalo, ni možno uporabiti statističnih meril. Predlagamo, da se k obstoječi skupini dodajo še naključno generirana vezja. V tem primeru lahko uporabimo tri nova merila: merilo uspešnosti, podobnosti in ekskluzivnosti. Dva simulatorja sta enako kakovostna, če imata enako uspešnost, podobnost in ekskluzivnost. Pokazali smo, da predlagana merila in postopek testiranja večata objektivnost primerjalnega testiranja.

Največji problem, ki nastopa pri testiranju, je določitev pričakovanih rezultatov (v našem primeru so to potenciali vseh vozlišč). Za vsako vezje bi bilo potrebno izračunati brez uporabe simulatorja delovno točko. Ker gre za naključna nelinearna vezja, ki niso vedno zaporedno-vzporedna, bi izračun delovne točke za npr. 1000 vezij predstavljal izredno naporno delo.

Ta problem bomo skušali rešiti z nadgradnjo obstoječega generatorja naključnih vezij, ki lahko tvori naključna vezja tako, da si najprej izbere naključno vrednost delovne točke in nato napravi sintezo vezja. Zaradi težav pri sintezi, zanekrat niso vključeni polprevodni elementi, ampak samo linearni R, I, V in nelinearni G in E.

8. Literatura

- /BRGLEZ,2000/ Franc Brglez "The Scientific Method and Design and Test", IEEE Design & Test of Computers, julij-september, 2000, str. 144,142,143.
- /CHUA,1975/ Leon O. Chua, Pen-Min Lin: "Computer-aided analysis of electronic circuits", Prentice-Hall, Englewood Cliffs, New Jersey, 1975.
- /CHUA,1976/ L. O. Chua and A. Ushida: "A switching-parameter algorithm for finding multiple solutions of nonlinear resistive circuits," IEEE Trans. Circuit Theory, Vol. CT-19, No. 4, str. 215-239, 1976.
- /DAVIDSON,2000/ S. Davidson, J. Harlow: "Guest Editors' Introduction: Benchmarking for Design and Test", IEEE Design & Test of Computers, julij-september, 2000, str. 12-14.
- /PERŠIČ,1995/ Boštjan Peršič: "Primer napačnega delovanja numerične integracije simulatorja SPICE", Elektrotehniški vestnik 62(2):117-125, 1995.
- /TRAJKOVIC,1991/ Lj. Trajkovic, R. C. Melville and S. C. Fang, "Finding dc operating points of transistor circuits using homotopy methods", Proc. IEEE Int. Symp. Circuits and Systems, Singapore, junij 1991, str. 758-761. (glej tudi <http://divine.eecs.berkeley.edu/~ljilja/>).

/TRAJKOVIC,1998/ Lj. Trajkovic, E. Fung, and S. Sanders, "HomSPICE: Simulator with homotopy algorithms for finding dc and steady state solutions of nonlinear circuits", Proc. IEEE Int. Symp. Circuits and Systems, Monterey, CA, June 1998, TPA 10-2.

Dr. Tomaž Dogša,
Fakulteta za elektrotehniko, računalništvo in
informatiko v Mariboru
Smetanova 17
2000 Maribor, Slovenija
Tel. (02) 220 7231
E-pošta: tdogsa@uni-mb.si

Prispelo (Arrived): 25.04.01

Sprejeto (Accepted): 01.06.01