

# Statička i dinamička neuronska mreža u identifikaciji sustava i procesa

---

**Pažanin, Ivan**

**Undergraduate thesis / Završni rad**

**2014**

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **University of Zagreb, Faculty of Mechanical Engineering and Naval Architecture / Sveučilište u Zagrebu, Fakultet strojarstva i brodogradnje**

*Permanent link / Trajna poveznica:* <https://urn.nsk.hr/urn:nbn:hr:235:961421>

*Rights / Prava:* [In copyright](#)/[Zaštićeno autorskim pravom.](#)

*Download date / Datum preuzimanja:* **2025-03-13**

*Repository / Repozitorij:*

[Repository of Faculty of Mechanical Engineering and Naval Architecture University of Zagreb](#)



SVEUČILIŠTE U ZAGREBU  
FAKULTET STROJARSTVA I BRODOGRANJE

# ZAVRŠNI RAD

**Ivan Pažanin**

Zagreb, 2014.

SVEUČILIŠTE U ZAGREBU  
FAKULTET STROJARSTVA I BRODOGRADNJE

# ZAVRŠNI RAD

Mentori:

Doc. dr. sc. Danko Brezak

Student:

Ivan Pažanin

Zagreb, 2014.

Izjavljujem da sam ovaj rad izradio samostalno koristeći stečena znanja tijekom studija i navedenu literaturu.

Ivan Pažanin



SVEUČILIŠTE U ZAGREBU  
**FAKULTET STROJARSTVA I BRODOGRADNJE**



Središnje povjerenstvo za završne i diplomske ispite  
Povjerenstvo za završne ispite studija strojarstva za smjerove:  
proizvodno inženjerstvo, računalno inženjerstvo, industrijsko inženjerstvo i menadžment, inženjerstvo  
materijala i mehatronika i robotika

Sveučilište u Zagrebu	
Fakultet strojarstva i brodogradnje	
Datum	Prilog
Klasa:	
Ur.broj:	

## ZAVRŠNI ZADATAK

Student: **Ivan Pažanin** Mat. br.: 0035169411

Naslov rada na hrvatskom jeziku: **Statička i dinamička neuronska mreža u identifikaciji sustava i procesa**

Naslov rada na engleskom jeziku: **Feed-forward and recurrent neural network in system/process identification**

Opis zadatka:

Umjetne neuronske mreže razvile su se sa ciljem brzog (paralelnog) procesiranja informacija, pokazujući pritom visoki stupanj adaptabilnosti, odnosno sposobnosti učenja različitih vrsta problema. Upotreba neuronskih mreža na području identifikacije dinamičkih sustava započinje početkom 90-ih godina prošlog stoljeća, i to najprije primjenom statičkih, a kasnije i dinamičkih mreža. Zahvaljujući svojim karakteristikama, algoritmi neuronskih mreža ubrzo su se pokazali kao iznimno dobar alat s velikom primjenom u modeliranju, posebice složenih, nelinearnih sustava i procesa.

Cilj ovog rada je usporedba i analiza karakteristika statičke i dinamičke neuronske mreže u problemima identifikacije linearnih i nelinearnih sustava/procesa.

U radu je potrebno:

1. Izraditi programsku podršku za troslojnu statičku i dinamičku neuronsku mrežu učenu primjenom EBP algoritma.
2. Provesti učenja obje mreže pod istim uvjetima na problemima identifikacije nekoliko različitih linearnih i nelinearnih sustava/procesa.
3. Generirati odzive mreža i usporediti dobivene rezultate.
4. Izvesti zaključke rada.

Zadatak zadan:

11. studenog 2013.

Zadatak zadao:

  
Doc. dr. sc. Danko Brezak

Rok predaje rada:

**1. rok:** 21. veljače 2014.

**2. rok:** 12. rujna 2014.

Predviđeni datumi obrane:

**1. rok:** 3., 4. i 5. ožujka 2014.

**2. rok:** 22., 23. i 24. rujna 2014.

Predsjednik Povjerenstva:

  
Prof. dr. sc. Zoran Kunica

## SADRŽAJ

SADRŽAJ .....	I
POPIS SLIKA .....	II
POPIS TABLICA.....	III
POPIS OZNAKA .....	IV
SAŽETAK.....	VI
SUMMARY .....	VII
1. Uvod .....	1
1.1. Biološki neuron .....	1
1.2. Umjetni neuron.....	2
2. Osnovne podjele umjetnih neuronskih mreža.....	3
3. Statička neuronska mreža .....	5
3.1. Model statičkog neurona .....	5
3.2. Model statičke neuronske mreže .....	6
3.3. Učenje mreže.....	7
3.3.1. Unaprijedna faza .....	7
3.3.2. Povratna faza.....	8
4. Dinamička neuronska mreža.....	12
4.1. Model dinamičkog neurona.....	12
4.2. Model dinamičke neuronske mreže .....	13
4.3. Učenje mreže.....	14
4.3.1. Unaprijedna faza .....	14
4.3.2. Povratna faza.....	15
5. Učenje, testiranje i usporedba.....	18
5.1. Linearni proporcionalni član prvog reda (P1).....	20
5.2. Nelinearni sustav .....	24
ZAKLJUČAK .....	28
LITERATURA.....	29

**POPIS SLIKA**

Slika 1.1	Biološki neuron .....	1
Slika 1.2	Umjetni neuron.....	2
Slika 2.1	Supervizorno učenje .....	4
Slika 3.1	Model statičkog neurona .....	5
Slika 3.2	Sigmoidalna funkcija.....	6
Slika 3.3	Model statičke neuronske mreže .....	7
Slika 4.1	Model dinamičkog neurona.....	12
Slika 4.2	Model dinamičke neuronske mreže.....	14
Slika 5.1	Princip učenja statičke (lijevo) i dinamičke (desno) mreže.....	18
Slika 5.2	Proces testiranja neuronskih mreža .....	19
Slika 5.3	Odziv P1 člana.....	20
Slika 5.4	Usporedba odziva statičke (lijevo) i dinamičke mreže (desno) pri učenju .....	21
Slika 5.5	Testovi statičke mreže s 2 skrivena neurona .....	22
Slika 5.6	Testovi dinamičke mreže s 5 skrivenih neurona .....	23
Slika 5.7	Odziv nelinearnog sustava.....	24
Slika 5.8	Usporedba odziva statičke (gore) i dinamičke mreže (dolje) pri učenju.....	25
Slika 5.9	Testovi statičke mreže s 10 skrivenih neurona.....	26
Slika 5.10	Testovi dinamičke mreže s 2 skrivena neurona.....	27

## **POPIS TABLICA**

Tablica 5.1 Usporedba algoritama pomoću NRMS mjere (linearni sustav) .....	21
Tablica 5.2 Usporedba algoritama pomoću NRMS mjere (nelinearni sustav).....	24



## POPIS OZNAKA

$A(z)$	nazivnik impulsne prijenosne funkcije dinamičkog neurona
$a_{ij}$	$i$ -ti koeficijent nazivnika prijenosne funkcije dinamičkog člana $j$ -tog dinamičkog neurona
$B(z)$	brojnik impulsne prijenosne funkcije dinamičkog neurona
$b_{ij}$	$i$ -ti koeficijent brojnika prijenosne funkcije dinamičkog člana $j$ -tog dinamičkog neurona
$BIAS$	neuron bez ulaza s konstantnom izlaznom vrijednosti jednakom 1
$D_{a_{ij}}$	parametar promjene $a_i$ koeficijenta $j$ -tog dinamičkog neurona
$D_{b_{ij}}$	parametar promjene $b_i$ koeficijenta $j$ -tog dinamičkog neurona
$D_{v_{ji}}$	parametar promjene težina između ulaznog $i$ skrivenog sloja
$d_k$	željena vrijednost $k$ -tog izlaza neuronske mreže
$E$	funkcija cilja (suma kvadrata pogrešaka)
$\Delta E$	promjena pogreške
$\nabla E$	gradijent pogreške
$I$	broj neurona ulaznog sloja mreže
$J$	broj neurona skrivenog sloja mreže
$K$	broj neurona izlaznog sloja mreže
$K_p$	pojačanje sustava
$NRMS$	normalizirani korijen srednje kvadratne pogreške – mjera točnosti
$net$	vrijednost funkcije sume
$net_{Hj}$	vrijednost funkcije sume $j$ -tog neurona skrivenog sloja
$net_{Hj}(n - i)$	vrijednost funkcije sume $j$ -tog neurona skrivenog sloja u $(n - i)$ koraku
$net_{Ok}$	vrijednost funkcije sume $k$ -tog neurona izlaznog sloja
$O_k$	$k$ -ti izlaz neuronske mreže
$u_j$	vrijednost $j$ -tog ulaza neurona
$v_{ji}$	težinski koeficijent između $j$ -tog neurona skrivenog sloja i $i$ -tog neurona ulaznog sloja
$w_j$	težinski koeficijent $j$ -tog ulaza neurona
$w_{kj}$	težinski koeficijent veze između $k$ -tog neurona izlaznog sloja i $j$ -tog neurona skrivenog sloja
$x$	vektor ulaznih vrijednosti (značajki) uzoraka skupa za učenje
$y$	vektor izlaza neurona
$\tilde{y}$	izlaz dinamičkog člana dinamičkog neurona
$\bar{y}$	ulaz aktivacijske funkcije dinamičkog neurona

$\tilde{y}(n - i)$	izlaz dinamičkog člana dinamičkog neurona u $(n - i)$ koraku
$y_j$	vrijednost izlaza $j$ -tog neurona skrivenog sloja
$Z_i$	$i$ -ti ulaz neuronske mreže
$z^{-i}[\text{net}(n)]$	operator linearnog pomaka
$\alpha$	vrijednost momentuma
$\gamma$	aktivacijska funkcija neurona
$\gamma'$	derivacija aktivacijske funkcije neurona
$\Delta$	iznos promjene težina u jednom koraku
$\delta$	parametar algoritma povratnog prostiranja pogreške
$\delta_{ok}$	parametar algoritma povratnog prostiranja pogreške izlaznog sloja
$\eta$	koeficijent brzine učenja
$\vartheta$	parametar učenja (težinski koeficijent, dinamički koeficijent)
$\vartheta(n)$	trenutna vrijednost parametra učenja
$\vartheta(n + 1)$	nova vrijednost parametra učenja
$\Delta\vartheta(n)$	tekuća promjena parametra učenja
$\sigma_{d_n}$	standardna devijacija željenih vrijednosti izlaza mreže

## **SAŽETAK**

Umjetna neuronska mreža predstavlja skup umjetnih neurona, temeljenih na građi i funkciji biološkog neurona, koji u međusobnoj interakciji vrše obradu ulaznog signala. Osim svojim mogućnostima paralelnog procesiranja informacija, umjetne neuronske mreže se odlikuju i svojim svojstvom učenja kroz niz ponavljajućih postupaka analize. Zahvaljujući ovoj karakteristici učenja, umjetne neuronske mreže pokazale su se veoma korisnima kod problema identifikacije dinamičkih sustava.

U ovom radu prikazuju se algoritmi jedne tipične statičke i jedne dinamičke mreže. Ocjenjuje se i uspoređuje kvaliteta učenja izrađenih statičkih i dinamičkih mreža, uz mijenjanje topologija, te se testiranjem razmatra njihova sposobnost generalizacije linearnog i nelinearnog sustava.

## **SUMMARY**

An artificial neural network is represented by an interconnected set of artificial neurons, which are based on biological neurons, that through mutual interaction process input data. Along with their ability to parallelly process data, artificial neural networks are characterized by their learning capabilities through repetitive data analysis. Due to this learning property, neural networks have proven to be very effective in the identification of dynamic systems.

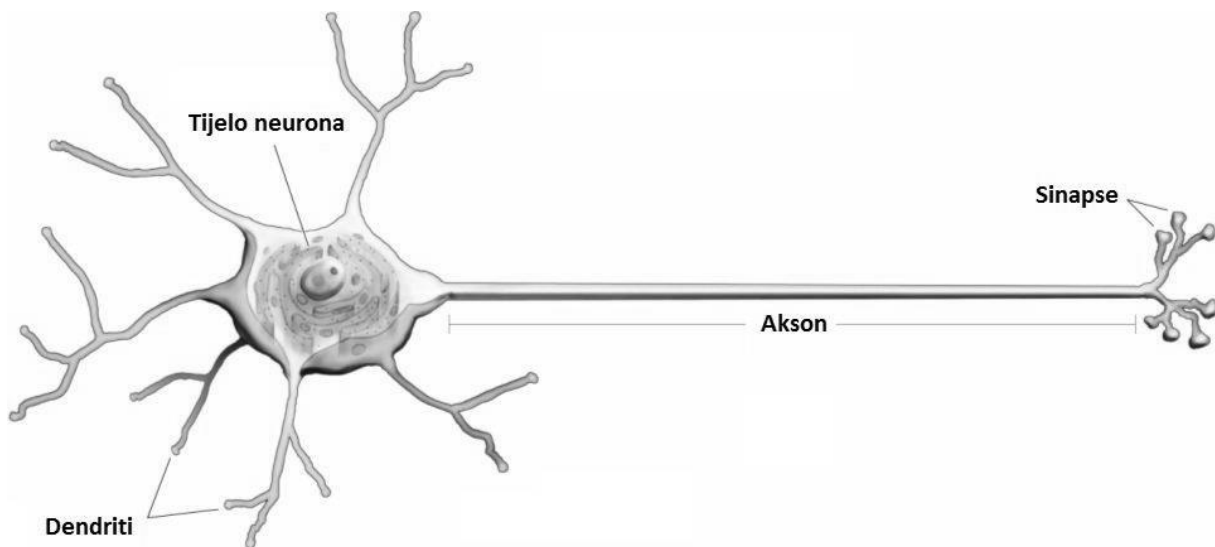
This paper covers typical static and dynamic neural network algorithms. While altering each network's topology, their learning characteristics are assessed and compared, as well as their generalizing aptitude of linear and nonlinear systems.

## 1. Uvod

### 1.1. Biološki neuron

Puno toga je i dalje nepoznato o kompleksnim procesima kojima mozak uči i procesira informacije. Osnovni model biološkog neurona predložio je William James još 1890. godine [1] tvrdnjom da aktivnost bilo koje točke mozga čovjeka predstavlja zbroj tendencija svih ostalih točaka da se u nju prazne.

Tipičan biološki neuron (Slika 1.1) pomoću dendrita prima signale susjednih neurona, a odašilje ih kroz svoj akson. Akson je dugačka cjevčica koja se na kraju dijeli na stotine grana, a na kraju svake grane nalazi se sinapsa. Akson jednog neurona može formirati sinaptičke veze s mnoštvom drugih neurona. Impulsi (izlazi) neurona putuju kroz akson do sinapsi odakle se signali različitih intenziteta (različito otežani signali) šalju kroz dendrite ili direktno na tijelo drugih neurona.



Slika 1.1 Biološki neuron

Neuron će poslati impuls kroz svoj akson ako je doveden u stanje dovoljne uzbuđene, pri čemu se uzima u obzir njegovo početno stanje uzbuđene i utjecaj signala ostalih neurona s kojima je također povezan. Signali koje neuron prima preko dendrita mogu biti inhibitorni ili uzbuđeni, matematički gledano suprotnog su predznaka. Da bi neuron poslao impuls kroz akson, njegova uzbuđena mora biti veća od smirujućeg utjecaja za kritični iznos, koji predstavlja prag osjetljivosti neurona.

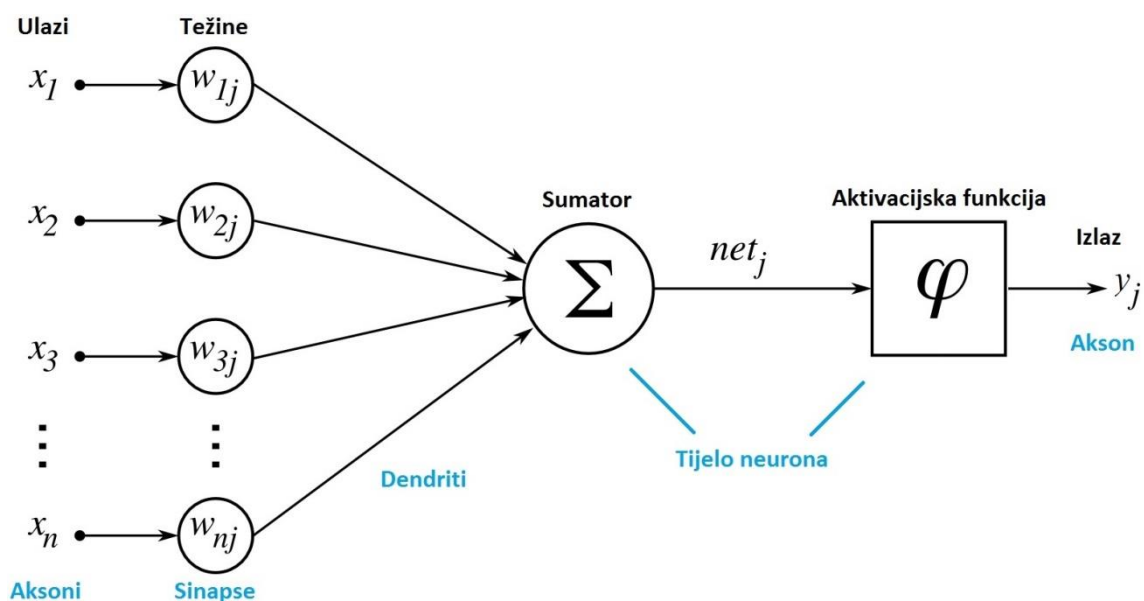
Mreža ovakvih neurona vrlo je složena i tako jedan neuron može generirati impuls koji će istovremeno smiriti ili uzbuđiti tisuće drugih neurona. Rezultat je visoki stupanj

povezanosti neurona u neuronskoj mreži, što upućuje na zaključak da se kompleksne funkcije neuronske mreže (misli, emocije, učenje...) prije ostvaruju kompleksnošću veza (težinama) među neuronima, nego isključivo kompleksnošću svakog pojedinog neurona.

## 1.2. Umjetni neuron

McCulloch i Pitts su 1943. godine [2] ponudili jednostavan model umjetnog neurona, baziranog na tvrdnji Williama Jamesa. Njihov model neurona se i danas koristi kao osnovni blok za izgradnju umjetnih neuronskih mreža.

Umjetni neuron (Slika 1.2) imitira osnovne principe rada biološkog neurona. Tijelo neurona može se podijeliti na dvije cjeline, sumator i aktivacijsku funkciju. Aktivacijska funkcija simulira prag osjetljivosti biološkog neurona i šalje signal na izlaz (akson), a pobudu prima preko sumatora. Sumator pomoću „dendrita“ dobavlja i zbraja inhibirajuće ili pobuđujuće signale od obližnjih neurona. Ulazi su zapravo izlazi susjednih neurona, koji preko svojih aksona šalju pobudu. Ulazi se množe s pozitivnim ili negativnim vrijednostima težina, čime se oponaša inhibirajući i pobudni učinak sinapsi.



Slika 1.2 Umjetni neuron

Naizgled umjetni neuron vjerno oponaša rad biološkog neurona, no ne uzima se u obzir utjecaj kašnjenja signala, što utječe na dinamiku sustava. Biološki neuron ne može zapremati jačine signala, što znači da u slučaju kašnjenja nekih signala neće biti zadovoljen prag osjetljivosti neurona te se neuron neće aktivirati. Stvara se efekt sinkronizacije, neuron će se aktivirati samo u slučaju istovremenog ispaljivanja susjednih neurona.

## 2. Osnovne podjele umjetnih neuronskih mreža

Postoje brojne konfiguracije umjetnih neuronskih mreža, a izbor konfiguracije ovisi o zadaći. Jedan sloj neuronske mreže složen je od skupa paralelno složenih neurona, pa se mreže dijele na:

- jednoslojne
- višeslojne

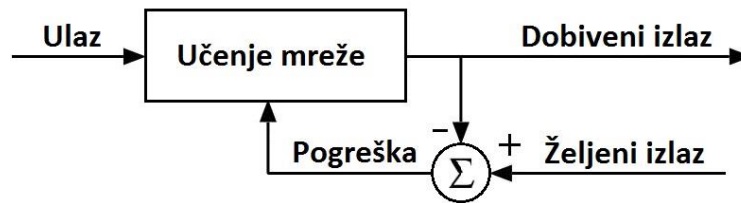
Višeslojne mreže obično imaju ulazni i izlazni sloj, a između njih jedan ili više skrivenih slojeva. Ako kroz slojeve signali putuju isključivo u jednom smjeru, unaprijed, tada se radi o unaprijednoj neuronskoj mreži. Ukoliko se barem jedan od signala vraća (povratna petlja), onda je riječ o povratnim neuronskim mrežama. Dakle, mreže se dalje dijele na:

- statičke (unaprijedne)
- dinamičke (povratne)

Neuronske mreže se kategoriziraju i prema načinu učenja:

- supervizorno (uz nadzor)
- nesupervizorno (bez nadzora)
- učenje s pojačanjem (nagradom)

Kod supervizornog učenja (Slika 2.1), vanjski učitelj prisutan je u procesu učenja mreže. On promatra ponašanje mreže te ju korigira dok se ne dobiju željeni rezultati. Kod ovog tipa učenja najprije se odredi struktura mreže (broj ulaza i izlaza, broj težina mreže, broj neurona te broj skrivenih slojeva). Vrijednosti težina se određuju generatorom slučajnih brojeva, a na ulaz se dovodi skup poznatih ulaznih varijabli. Mreža sa zadanim vrijednostima producira skup izlaznih varijabli koje se uskoređuju sa željenim izlazima. Preko razlike željenih i dobivenih vrijednosti određuje se razina pogreške mreže, koja se koristi za računanje novih težina pomoću nekog od algoritama. Cijeli postupak se ponavlja iteracijski dok se ne postigne željena točnost mreže. Nakon učenja mreže slijedi testiranje, učitavaju se nove vrijednosti ulaza (na kojima mreža nije učila) i uspoređuju s novim vrijednostima izlaza. Iznos pogreške govori o robusnosti mreže, tj. ukazuje koliko je mreža sposobna davati pogodne rezultate za skup ulaza kojim nije bila učena, što je bitno za primjenu mreže kada izlazi neće biti poznati.



Slika 2.1 Supervizorno učenje

Nesupervizorne mreže ne koriste vanjskog učitelja, na ulaz mreže se dovodi skup ulaznih varijabli, a mreža se samoorganizira podešavanjem svojih težina. Jer željeni izlaz mreže nije određen, rezultat učenja je nepredvidiv.

Kod učenja s pojačanjem neuronska mreža dobija smjernice u obliku nagrada, ispravno ponašanje se nagrađuje, dok se neispravno ne korigira. Razlikuje se od supervizornog učenja jer su mreži ulazi i izlazi nepoznati, ulaze mreža sama pribavlja iz okoline.

Bitno za dinamiku sustava je i izbor aktivacijske funkcije, a one mogu biti:

- linearne
- nelinearne

Linearne aktivacijske funkcije množe izlaz iz sumatora nekim konstantnim faktorom i rezultat se šalje na izlaz. Najčešće nelinearne funkcije su funkcije praga osjetljivosti, sigmoidalne, hiperboličke i harmoničke.

Mreže se razlikuju i prema algoritmu učenja, kojim se podešavaju parametri (težine) mreže. Algoritmi su se kroz dugi niz godina stvarali i poboljšavali, tako danas postoji veliki izbor različitih algoritama. Neki od poznatijih uključuju:

- generalizirano delta pravilo
- povratno propagiranje
- Grossbergovo (Hebbianovo) učenje
- primjena Lyapunovih funkcija
- optimalno učenje
- statističko učenje
- adaptivno učenje
- ...

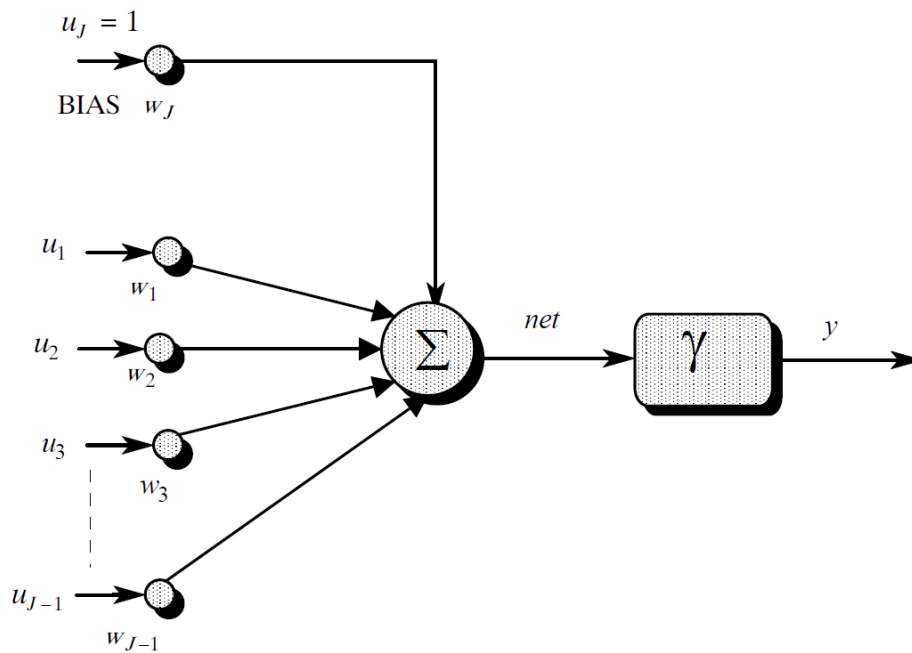
U nastavku se detaljnije opisuju i koriste statička i dinamička troslojna neuronska mreža. U oba slučaja koristi se nelinearna sigmoidalna aktivacijska funkcija te supervizorno učenje povratnim propagiranjem pogreške.



### 3. Statička neuronska mreža

#### 3.1. Model statičkog neurona

Standardni model statičkog neurona, prikazan Slika 3.1, sastoji se od funkcije sume  $\Sigma$  i aktivacijske funkcije  $\gamma$ . Osnovna značajka svakog statičkog neurona je da posjeduje više ulaza i samo jedan izlaz. Također, svakom neuronu dodan je nepromjenjivi ulaz jedinične vrijednosti (Bias).



Slika 3.1 Model statičkog neurona

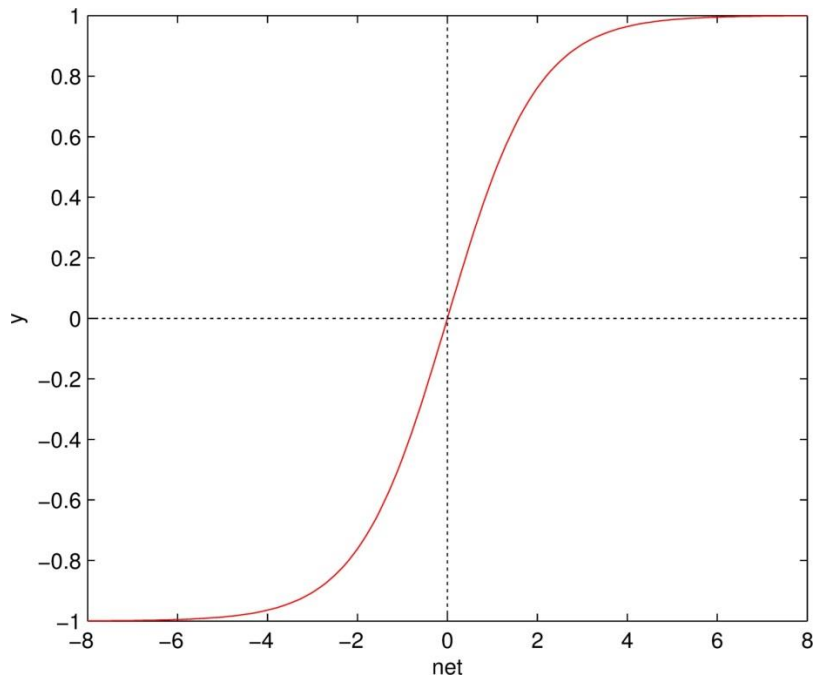
Sumator zbraja ulaze pomnožene s njihovim težinama prema izrazu (3-1), rezultat **net** se preslikava u izlaz **y** preko aktivacijske funkcije po izrazu (3-2).

$$net = \sum_{j=1}^J w_j u_j , \quad (3-1)$$

$$y = \gamma(net) . \quad (3-2)$$

Aktivacijska funkcija obično je neka monotono rastuća funkcija sa zasićenjem. Odabrana sigmoidalna aktivacijska funkcija (Slika 3.2) normira izlaze na raspon [-1,1] relacijom (3-3):

$$y = \frac{2}{1 + e^{-net}} - 1 . \quad (3-3)$$



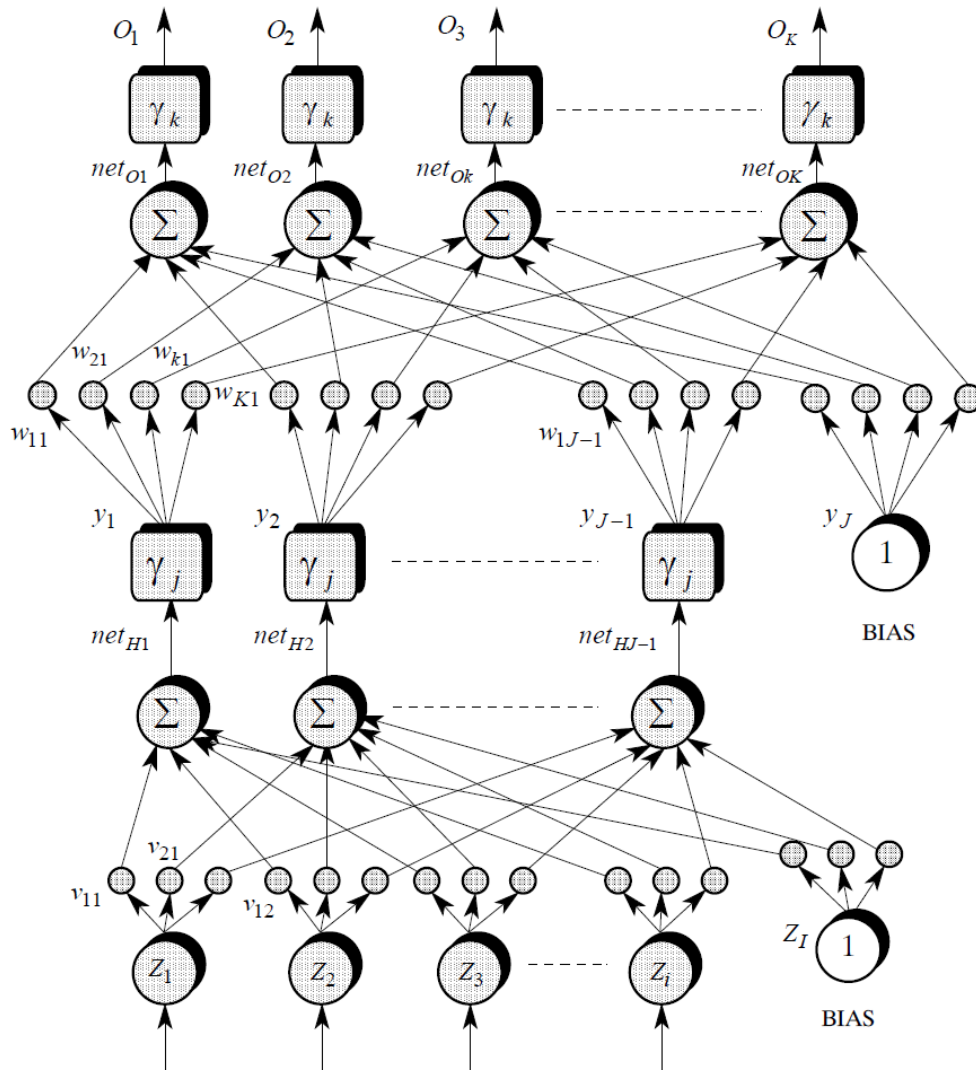
**Slika 3.2 Sigmoidalna funkcija**

Zbog numeričke stabilnost mreže potrebno je mrežu učiti i testirati normiranim vrijednostima, svi ulazi i izlazi se normiraju na raspon  $[-1,1]$ .

### 3.2. Model statičke neuronske mreže

Statička unaprijedna višeslojna mreža sastoji se od neurona povezanih u paralelne slojeve, Slika 3.3 prikazan je model troslojne mreže. Ulazni i izlazni sloj u direktnoj su interakciji s okolinom, dok skriveni nije. Neuroni u ulaznom sloju nisu pravi neuroni, nego čvorovi za distribuciju ulaza neuronske mreže prema prvom skrivenom sloju neurona. Parametri mreže su:

- $Z_i$  – ulazi u mrežu
- $v_{ji}$  – težine veza između ulaznog i skrivenog sloja
- $w_{kj}$  – težine veza između skrivenog i izlaznog sloja
- $O_k$  – izlazi iz mreže
- $BIAS$  – neuroni izlaza jedinične vrijednosti



Slika 3.3 Model statičke neuronske mreže

### 3.3. Učenje mreže

Postupak učenja može se podijeliti na dvije faze:

1. unaprijedna faza
2. povratna faza

#### 3.3.1. Unaprijedna faza

U unaprijednoj fazi učenja učitavaju se vrijednosti ulaza mreže  $Z$  iz skupa za učenje te se izračunavaju vrijednosti izlaza mreže  $O$ . Dobivene vrijednosti uspoređuju se s traženim vrijednostima izlaza  $d$ . Za početak rada mreže potrebno je zadati i početne vrijednosti težina  $W$  i  $V$ , koje se zadaju generatorom slučajnih brojeva. Radi stabilnog rada mreže, sve vrijednosti moraju biti normirane na raspon  $[-1 \ 1]$ .

Opisan je jedan prolaz kroz unaprijednu fazu:

$$net_{Hj} = \sum_{i=1}^I v_{ji} z_i, \quad j = 1, 2, \dots, J-1, \quad i = 1, 2, \dots, I, \quad (3-4)$$

$H$  – oznaka sloja,  
 $I$  – broj ulaznih neurona + 1  
 $J$  – broj neurona skrivenog sloja

$$y_j = \frac{2}{1 + e^{-net_{Hj}}} - 1, \quad j = 1, 2, \dots, J-1, \quad (3-5)$$

$$y_J = 1, \quad Bias, \quad (3-6)$$

$$net_{Ok} = \sum_{j=1}^J w_{kj} y_j, \quad k = 1, 2, \dots, K, \quad (3-7)$$

$K$  – broj neurona izlaznog sloja (broj izlaza)

$$O_k = K_p net_{Ok}, \quad k = 1, 2, \dots, K. \quad (3-8)$$

$K_p$  – nagib linearne aktivacijske funkcije (u nastavku  $K_p = 1$ )

### 3.3.2. Povratna faza

U povratnoj fazi izračunava se pogreška mreže uspoređivanjem tražene vrijednosti izlaza  $\mathbf{d}$  s dobivenim izlazima  $\mathbf{O}$ . Podešavaju se vrijednosti  $\mathbf{W}$  i  $\mathbf{V}$  te se iznova izvršava unaprijedna faza sve dok se ne postigne zadovoljavajuće niska razina pogreške.

Uobičajena statistička metoda regresijske analize, suma kvadrata pogreške kao mjera odstupanja izlaza mreže od željene vrijednosti izlaza, najčešće je korištena funkcija cilja [3]:

$$E = \frac{1}{2} \sum_{n=1}^N (d_n - O_n)^2, \quad (3-9)$$

$N$  – broj elemenata u skupu za učenje

$$\vartheta(n+1) = \vartheta(n) + \Delta\vartheta(n). \quad (3-10)$$

$n$  – trenutni korak učenja

$\Delta\vartheta(n)$  – veličina promjene parametara učenja

$\vartheta(n+1)$  – nova vrijednost parametra učenja

Za izlazni sloj  $\vartheta = \mathbf{W}$ , a za skriveni sloj  $\vartheta = \mathbf{V}$ .

Algoritam povratnog prostiranja pogreške (eng. Error Back-Propagation Algorithm) jednostavna je metoda za izračunavanje gradijenta pogreške s obzirom na parametre učenja. Promjena parametara učenja odvija se u suprotnom smjeru od toka signala.

Pogrešku  $E(\vartheta)$  moguće je u okolišu točke  $\vartheta$  aproksimirati s prva dva člana Taylorovog reda:

$$E(\vartheta + \Delta\vartheta) \approx E(\vartheta) + \Delta E(\vartheta), \quad (3-11)$$

$$\Delta E(\vartheta) = \Delta\vartheta^T \nabla E(\vartheta), \quad (3-12)$$

$$\nabla E(\vartheta) = \frac{\partial E(\vartheta)}{\partial \vartheta}. \quad (3-13)$$

Izraz (3-13) naziva se gradijentom pogreške. Da bi se pogreška smanjivala najvećim mogućim iznosom, treba odrediti  $\Delta\vartheta$  za koji promjena pogreške učenja  $\Delta E(\vartheta)$  poprima najveći negativni iznos, a to se ostvaruje uz uvjet:

$$\Delta\vartheta = -\eta\nabla E(\vartheta) . \quad (3-14)$$

$\eta$  – koeficijent brzine učenja (mjera promjene)

Koeficijent brzine učenja  $\eta$  određuje učitelj, a pokazalo se da je njegova vrijednost najčešće između  $10^{-3}$  i 10 [3]. Izraz (3-14) osigurava promjenu težinskih koeficijenata u smjeru najstrmijeg pada ukupne pogreške. Uvrštavanjem (3-14) u (3-10) dobiva se algoritam za izračunavanje promjena težina mreže:

$$\vartheta(n+1) = \vartheta(n) - \eta\nabla E(\vartheta(n)) . \quad (3-15)$$

Za učenje EBP algoritmom potreban je veliki broj iteracija da bi se greška smanjila na željenu vrijednost. Da bi se umanjio taj problem i ostali problemi tipični za EBP algoritam, radi se modifikacija algoritma dodavanjem momentuma prema:

$$\Delta\vartheta = -\eta\nabla E(\vartheta) + \alpha\Delta\vartheta(n-1) . \quad (3-16)$$

$\alpha$  – koeficijent momentuma

Vrijednost koeficijenta momentuma  $\alpha$  obično je u rasponu od 0.1 i 0.9 [3]. Momentum skalira promjenu parametara učenja, u prethodnom koraku se dodaje u izraz za promjenu u sadašnjem koraku:

$$\vartheta(n+1) = \vartheta(n) - \eta\nabla E(\vartheta(n)) + \alpha\Delta\vartheta(n-1) . \quad (3-17)$$

Slijedi promjena parametara učenja od izlaznog prema skrivenom sloju. Prvo se mijenjaju težinski faktori između izlaznog i skrivenog sloja  $W$ :

$$W_{kj}(n+1) = W_{kj}(n) - \eta\nabla E(n) + \alpha\Delta W_{kj}(n-1) . \quad (3-18)$$

Izraz za gradijent pogreške:

$$\nabla E(n) = \frac{\partial E(n)}{\partial W_{kj}} = \frac{\partial E(n)}{\partial O_k} \frac{\partial O_k}{\partial net_{Ok}} \frac{\partial net_{Ok}}{\partial W_{kj}} . \quad (3-19)$$

Uvodi se karakteristična vrijednost algoritma povratnog prostiranja pogreške  $\delta$ :

$$\delta = \frac{\partial E(n)}{\partial net} . \quad (3-20)$$

Iz (3-9) slijedi:

$$\frac{\partial E(n)}{\partial O_k} = -(d_k - O_k) . \quad (3-21)$$

Iz (3-8) slijedi:

$$\frac{\partial O_k}{\partial net_{Ok}} = 1 . \quad (3-22)$$

Iz (3-19) do (3-22) slijedi izraz za  $\delta$ :

$$\delta_{Ok} = d_k - O_k . \quad (3-23)$$

Prema (3-7) može se odrediti i posljednja parcijalna derivacija izraza (3-19):

$$\frac{\partial net_{Ok}}{\partial W_{kj}} = y_j . \quad (3-24)$$

Uvrštavanjem izraza (3-21), (3-22), (3-23) u (3-19) slijedi:

$$\nabla E(n) = \frac{\partial E(n)}{\partial W_{kj}} = -(d_k - O_k)y_j = \delta_{Ok}y_j . \quad (3-25)$$

Dobija se konačni algoritam promjene težina izlaznog sloja:

$$W_{kj}(n+1) = W_{kj}(n) + \eta \delta_{Ok}y_j + \alpha \Delta W_{kj}(n-1) . \quad (3-26)$$

Nakon promjene težina izlaznog sloja, slijedi promjena težina skrivenog sloja, temeljna jednačba prema kojoj se vrši adaptacija težina skrivenog sloja u svojoj se biti ne razlikuje od (3-18).

$$V_{ji}(n+1) = V_{ji}(n) - \eta \nabla E(n) + \alpha \Delta V_{ji}(n-1) . \quad (3-27)$$

Slijedi izračun gradijenta pogreške:

$$\nabla E(n) = \frac{\partial E(n)}{\partial V_{ji}} = \frac{\partial E(n)}{\partial y_j} \frac{\partial y_j}{\partial net_{Hj}} \frac{\partial net_{Hj}}{\partial V_{ji}} . \quad (3-28)$$

Na promjenu svake težine skrivenog sloja utječu svi neuroni izlaznog sloja, dakle:

$$\frac{\partial E(n)}{\partial y_j} = \sum_{k=1}^K \frac{\partial E(n)}{\partial O_k} \frac{\partial O_k}{\partial net_{Ok}} \frac{\partial net_{Ok}}{\partial y_j} . \quad (3-29)$$

Pri tome je:

$$\frac{\partial E(n)}{\partial O_k} = -(d_k - O_k), \quad k = 1, 2, \dots, K , \quad (3-30)$$

$$\frac{\partial O_k}{\partial net_{Ok}} = 1, \quad k = 1, 2, \dots, K , \quad (3-31)$$

$$\frac{\partial net_{Ok}}{\partial y_j} = W_{kj}, \quad k = 1, 2, \dots, K, \quad j = 1, 2, \dots, J - 1 . \quad (3-32)$$

Uvrštavanjem izraza (3-30), (3-31) i (3-32) u (3-29) dobiva se:

$$\frac{\partial E(n)}{\partial y_j} = - \sum_{k=1}^K (d_k - O_k) W_{kj} . \quad (3-33)$$

Uvrštavanjem (3-23) u (3-33) dobija se konačni oblik rješenja prvog razlomka na desnoj strani jednakosti (3-28):

$$\frac{\partial E(n)}{\partial y_j} = - \sum_{k=1}^K \delta_{Ok} W_{kj} . \quad (3-34)$$

A Drugi i treći razlomak određuju se iz izraza (3-5) i (3-6):

$$\frac{\partial y_j}{\partial \text{net}_{Hj}} = \gamma_j' = \frac{1}{2}(1 - y_j^2) , \quad (3-35)$$

$$\frac{\partial \text{net}_{Hj}}{\partial V_{ji}} = Z_i . \quad (3-36)$$

Konačni oblik algoritma promjene težina skrivenog sloja:

$$V_{ji}(n + 1) = V_{ji}(n) + \frac{1}{2}\eta(1 - y_j^2)Z_i \left( \sum_{k=q}^K \delta_{Ok} W_{kj} \right) + \alpha \Delta V_{ji}(n - 1) . \quad (3-37)$$





Vrijednosti izlaza dinamičkog člana pribraja se težina veze s Bias neuronom:

$$\bar{y}(n) = \tilde{y}(n) + W_j . \quad (4-5)$$

Rezultat se obrađuje istom nelinearnom sigmoidalnom aktivacijskom funkcijom iz primjera statičke neuronske mreže i glasi:

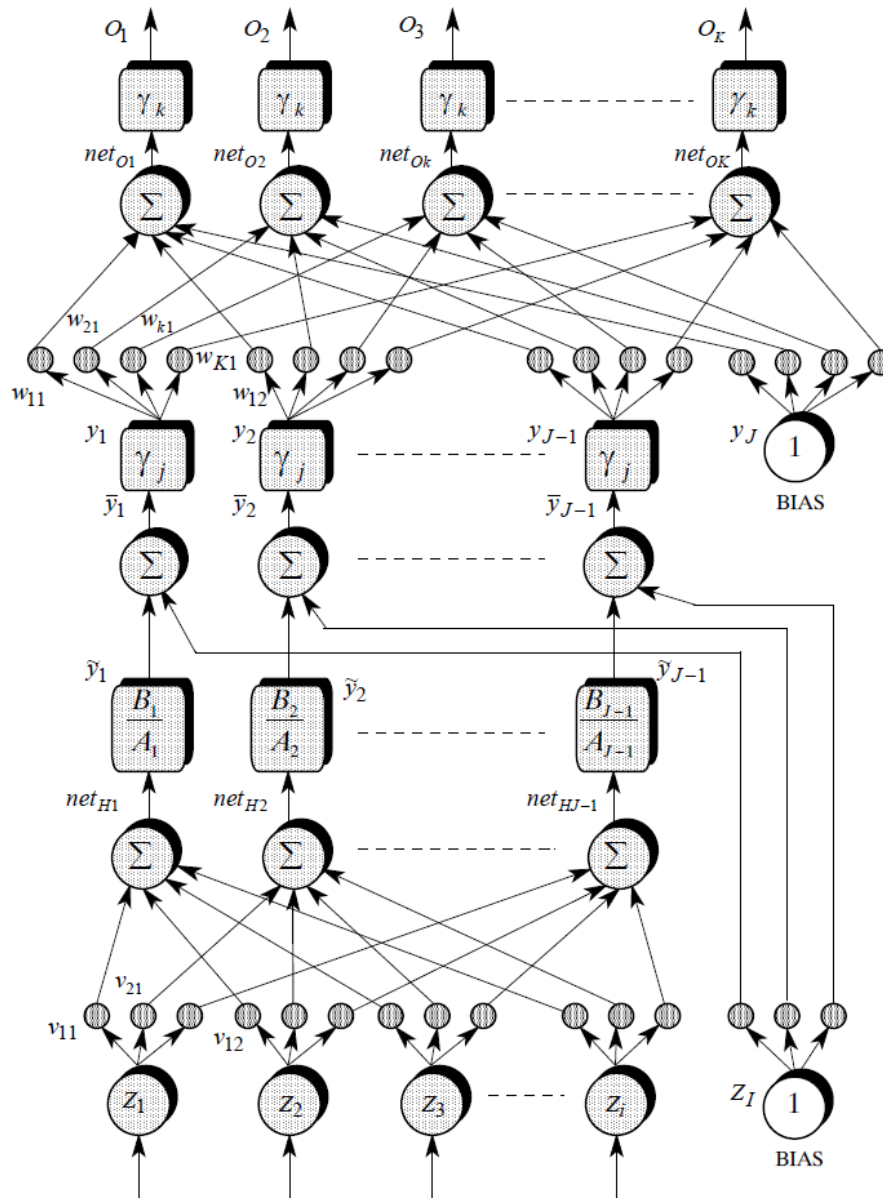
$$y(n) = \gamma(\bar{y}(n)) = \frac{2}{1 + e^{-\bar{y}(n)}} - 1 . \quad (4-6)$$

#### 4.2. Model dinamičke neuronske mreže

Kao i kod statičke neuronske mreže, prikazana je troslojna neuronska mreža sastavljena od ulaznog, skrivenog i izlaznog sloja (Slika 4.2). Ulazni sloj predstavljaju čvorovi za distribuciju ulaza skrivenom sloju neuronske mreže. Skriveni sloj građen je od dinamičkog neurona, a izlaz od prije opisanog običnog statičkog neurona (Slika 3.1). Razlog korištenja dinamičkih neurona samo u skrivenom sloju mreže je pojednostavljenje kompleksnosti proračuna, a pokazalo se da je doprinos kvaliteti učenja mreže zanemariv [3].

Parametri mreže su:

- $Z_i$  – ulazi u mrežu
- $v_{ji}$  – težine veza između ulaznog i skrivenog sloja
- $a_{ij}$  – koeficijent nazivnika prijenosne funkcije dinamičkog člana
- $b_{ij}$  – koeficijent brojnika prijenosne funkcije dinamičkog člana
- $w_{kj}$  – težine veza između skrivenog i izlaznog sloja
- $O_k$  – izlazi iz mreže
- $BIAS$  – neuroni izlaza jedinične vrijednosti



Slika 4.2 Model dinamičke neuronske mreže

### 4.3. Učenje mreže

Učenje dinamičke neuronske mreže se također dijeli u dvije cjeline:

1. unaprijedna faza
2. povratna faza

#### 4.3.1. Unaprijedna faza

Učitavaju se vrijednosti ulaza mreže  $Z$  iz skupa za učenje te se izračunavaju vrijednosti izlaza mreže  $O$ . Izračunate izlazne vrijednosti uspoređuju se s traženim vrijednostima izlaza  $d$ . Kod dinamičke mreže je uz koeficijente težina  $W$  i  $V$  potrebno zadati parametre  $A$  i  $B$ . Radi stabilnog rada mreže, sve vrijednosti se normiraju na raspon  $[-1 \ 1]$ .

Prolaz kroz unaprijednu fazu slični na prolaz kroz unaprijednu fazu statičke mreže, a izgleda ovako:

$$net_{Hj} = \sum_{i=1}^{I-1} V_{ji}Z_i, \quad j = 1, 2, \dots, J-1, \quad i = 1, 2, \dots, I-1, \quad (4-7)$$

$$\tilde{y}(n) = b_{0j}(n)net_{Hj}(n) + b_{1j}(n)net_{Hj}(n-1) + b_{2j}(n)net_{Hj}(n-2) - a_{1j}(n)\tilde{y}(n-1) - a_{2j}(n)\tilde{y}(n-2), \quad (4-8)$$

$$\bar{y}_j(n) = \tilde{y}_j(n) + V_{jI}(n)Z_I, \quad (4-9)$$

gdje je  $Z_I = 1$  (Bias).

$$y_j(n) = \frac{2}{1 + e^{-\bar{y}_j(n)}} - 1, \quad j = 1, 2, \dots, J-1, \quad (4-10)$$

$$y_J = 1, \quad \text{Bias}, \quad (4-11)$$

$$net_{Ok}(n) = \sum_{j=1}^J y_j(n)W_{kj}(n), \quad (4-12)$$

$$O_k(n) = net_{Ok}(n), \quad k = 1, 2, \dots, K. \quad (4-13)$$

#### 4.3.2. Povratna faza

Ponovno se koristi suma kvadrata pogreške kao mjera odstupanja:

$$E = \frac{1}{2} \sum_{n=1}^N (d_n - O_n)^2. \quad (4-14)$$

Promjena parametara učenja, težinskih i dinamičkih koeficijenata, odvija se prema:

$$\vartheta(n+1) = \vartheta(n) - \eta \nabla E(\vartheta(n)) - \alpha \Delta \vartheta(n-1). \quad (4-15)$$

Promjena težinskih koeficijenata veza između izlaznog i skrivenog sloja:

$$W_{kj}(n+1) = W_{kj}(n) - \eta \nabla E(n) - \alpha \Delta W_{kj}(n-1). \quad (4-16)$$

Slijedi promjena koeficijenata dinamičkih članova skrivenog sloja, impulsnoj funkciji danoj izrazom (4-2) definiran je linearni operator vremenskog pomaka:

$$\frac{[\tilde{y}(n)]}{[net(n)]} = \frac{B(z)}{A(z)}. \quad (4-17)$$

Iz čega je:

$$A(z)[\tilde{y}(n)] = \tilde{y}(n) + a_1\tilde{y}(n-1) + a_2\tilde{y}(n-2), \quad (4-18)$$

$$B(z)[net(n)] = b_0net(n) + b_1net(n-1) + b_2net(n-2). \quad (4-19)$$

Da bi se korigirala vrijednost parametra  $\mathbf{b}$ , potrebno je izračunati gradijent pogreške  $\nabla E(n)$  koji glasi:

$$\nabla E(n) = \frac{\partial E(n)}{\partial b_{ij}} = \frac{\partial E}{\partial y_j} \frac{\partial y_j}{\partial \bar{y}_j} \frac{\partial \bar{y}_j}{\partial \tilde{y}_j} \frac{\partial \tilde{y}_j}{\partial b_{ij}}. \quad (4-20)$$

Prvi razlomak izraza (4-20) izveden je izrazom (3-34), a glasi:

$$\frac{\partial E(n)}{\partial y_j} = - \sum_{k=1}^K \delta_{Ok} W_{kj} . \quad (4-21)$$

Ostali razlomci:

$$\frac{\partial y_j}{\partial \bar{y}_j} = \gamma_j' = \frac{1}{2} (1 - y_j^2) , \quad (4-22)$$

$$\frac{\partial \bar{y}_j}{\partial \tilde{y}_j} = 1 . \quad (4-23)$$

Te za zadnji razlomak, primjenom pokazanog operatora vremenskog pomaka izvodi se:

$$\left. \frac{\partial [\tilde{y}(n)]}{\partial \vartheta} \right|_{\vartheta=b_{ij}} = [D_{b_{ij}}(n)] = \frac{z^{-1}}{A(z)} [net(n)], \quad i = 0,1,2 , \quad (4-24)$$

$$D_{b_{ij}}(n) = net_j(n-i) - a_{1j} D_{b_{ij}}(n-1) - a_{2j} D_{b_{ij}}(n-2), \quad i = 0,1,2 . \quad (4-25)$$

Uvrštavanjem izraza (4-21), (4-22), (4-23) i (4-25) u izraz (4-20) dobiva se:

$$\nabla E(n) = -\frac{1}{2} (1 - y_j^2) D_{b_{ij}}(n) \sum_{k=1}^K \delta_{Ok} W_{kj}, \quad i = 0,1,2 . \quad (4-26)$$

Tako da konačni oblik algoritma promjene parametra učenja **B** poprima oblik:

$$b_{ij}(n+1) = b_{ij}(n) + \frac{1}{2} \eta (1 - y_j^2) D_{b_{ij}}(n) \left( \sum_{k=1}^K \delta_{Ok} W_{kj} \right) - \alpha \Delta b_{ij}(n-1), \quad i = 0,1,2 . \quad (4-27)$$

Parametar **A** izvodi se na isti način, jedina razlika je u parcijalnoj derivaciji zadnjeg razlomka izraza (4-20), rezultat je:

$$\left. \frac{\partial [\tilde{y}(n)]}{\partial \vartheta} \right|_{\vartheta=a_{ij}} = [D_{a_{ij}}(n)] = \frac{-z^{-1}}{A(z)} [\tilde{y}(n)], \quad i = 1,2 , \quad (4-28)$$

$$D_{a_{ij}}(n) = -\tilde{y}_j(n-i) - a_{1j} D_{a_{ij}}(n-1) - a_{2j} D_{a_{ij}}(n-2), \quad i = 1,2 , \quad (4-29)$$

$$\nabla E(n) = -\frac{1}{2} (1 - y_j^2) D_{a_{ij}}(n) \sum_{k=1}^K \delta_{Ok} W_{kj}, \quad i = 1,2 . \quad (4-30)$$

Konačni oblik algoritma promjene parametra učenja **A** je dakle:

$$a_{ij}(n+1) = a_{ij}(n) + \frac{1}{2} \eta (1 - y_j^2) D_{a_{ij}}(n) \left( \sum_{k=1}^K \delta_{Ok} W_{kj} \right) - \alpha \Delta a_{ij}(n-1), \quad i = 1,2 . \quad (4-31)$$

Preostaje još odrediti algoritam promjene težinskih koeficijenata veza između skrivenog i ulaznog sloja neurona, **V**. Ponovno se mijenja parcijalna derivacija zadnjeg razlomka izraza (4-20):

$$\left. \frac{\partial [\tilde{y}(n)]}{\partial \vartheta} \right|_{\vartheta=v_{ji}} = [D_{V_{ij}}(n)] = \frac{B(z)}{A(z)} [Z_i(n)] , \quad (4-32)$$

$$D_{V_{ji}} = b_{0j}Z_i(n) + b_{1j}Z_i(n-1) + b_{2j}Z_i(n-2) - a_{1j}D_{V_{ji}}(n-1) - a_{2j}D_{V_{ji}}(n-2) , \quad (4-33)$$

$$\nabla E(n) = -\frac{1}{2}(1-y_j^2)D_{V_{ji}}(n) \sum_{k=1}^K \delta_{Ok}W_{kj} . \quad (4-34)$$

Slijedi:

$$V_{ji}(n+1) = V_{ji}(n) + \frac{1}{2}\eta(1-y_j^2)D_{V_{ji}}(n) \left( \sum_{k=1}^K \delta_{Ok}W_{kj} \right) - \alpha\Delta V_{ji}(n-1) . \quad (4-35)$$

Zadnja ostaje jedino težina pridružena Bias neuronu ulaznog sloja, kao što se vidi iz slike [Slika 4.2].

$$\nabla E(n) = \frac{\partial E(n)}{\partial V_{jl}} = \frac{\partial E}{\partial y_j} \frac{\partial y_j}{\partial \bar{y}_j} \frac{\partial \bar{y}_j}{\partial V_{jl}} . \quad (4-36)$$

Prva dva razlomka dana su izrazima (4-22) i (4-21), a treći je jednak:

$$\frac{\partial \bar{y}_j}{\partial V_{jl}} = Z_l = 1 , \quad (4-37)$$

$$\nabla E(n) = -\frac{1}{2}(1-y_j^2) \sum_{k=1}^K \delta_{Ok}W_{kj} . \quad (4-38)$$

Dakle dobiva se:

$$V_{jl}(n+1) = V_{jl}(n) + \frac{1}{2}\eta(1-y_j^2) \left( \sum_{k=1}^K \delta_{Ok}W_{kj} \right) - \alpha\Delta V_{jl}(n-1) . \quad (4-39)$$

## 5. Učenje, testiranje i usporedba

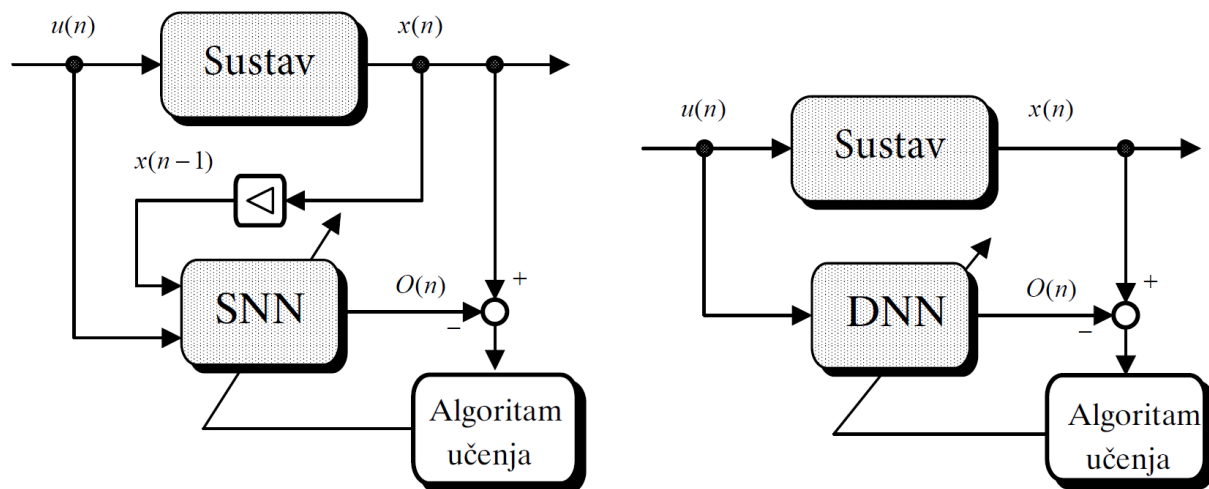
Dvije navedene neuronske mreže, statička i dinamička, učene su istim podacima i to za dva različita slučaja. Cilj je naučiti svaku mrežu da što vjernije oponaša dinamiku sustava. Prvi primjer je jednostavan linearni dinamički član prvog reda (P1 član). A drugi, nelinearni sustav, može se opisati jednadžbom diferencije prvog reda uz period uzorkovanja od 1 sekunde i vremenske konstante od 10 sekundi:

$$x(n+1) = (0.9 - 0.003x(n))x(n) + 0.2u(n) . \quad (5-1)$$

Radi usporedbe, statička i dinamička neuronska mreža testirana je s 2, 5 i 10 neurona skrivenog sloja. Ostali parametri učenja su:

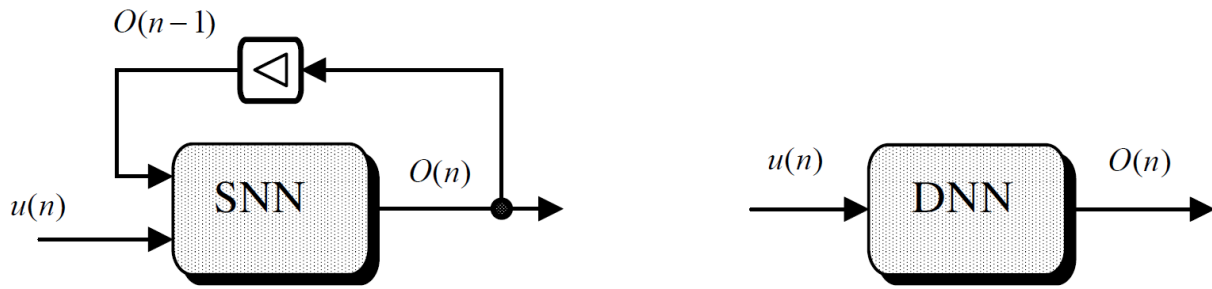
- $\eta = 0.01$  – koeficijent brzine učenja
- $\alpha = 0.8$  – vrijednost momentuma

Postupak učenja mreža shematski je prikazan Slika 5.1. Vidljivo je iz slike da se statičkoj mreži, u fazi učenja, vraća izlaz skupa  $\mathbf{d}$  u obliku dodatnog ulaza i to s kašnjenjem od  $(n - 1)$ . Dakle, sve statičke neuronske mreže imaju dva ulaza dok dinamičke imaju samo jedan.



**Slika 5.1** Princip učenja statičke (lijevo) i dinamičke (desno) mreže

Analogno tome, i pri testiranju se statičkoj neuronskoj mreži vraćaju izlazne vrijednosti kao dodatni ulazi. Mreža u praktičnoj primjeni tokom rada ne može znati točnu (izmjerenu) vrijednost izlaza, tako da se povratna veza ostvaruje s izračunatim vrijednostima prethodnog koraka (Slika 5.2).



Slika 5.2 Proces testiranja neuronskih mreža

Za procjenu uspješnosti algoritama učenja u rješavanju postavljenog zadatka, te njihovu usporedbu, koristi se NRMS<sup>1</sup> mjera točnosti:

$$\bar{d} = \frac{1}{N} \sum_{n=1}^N d_n, \quad (5-2)$$

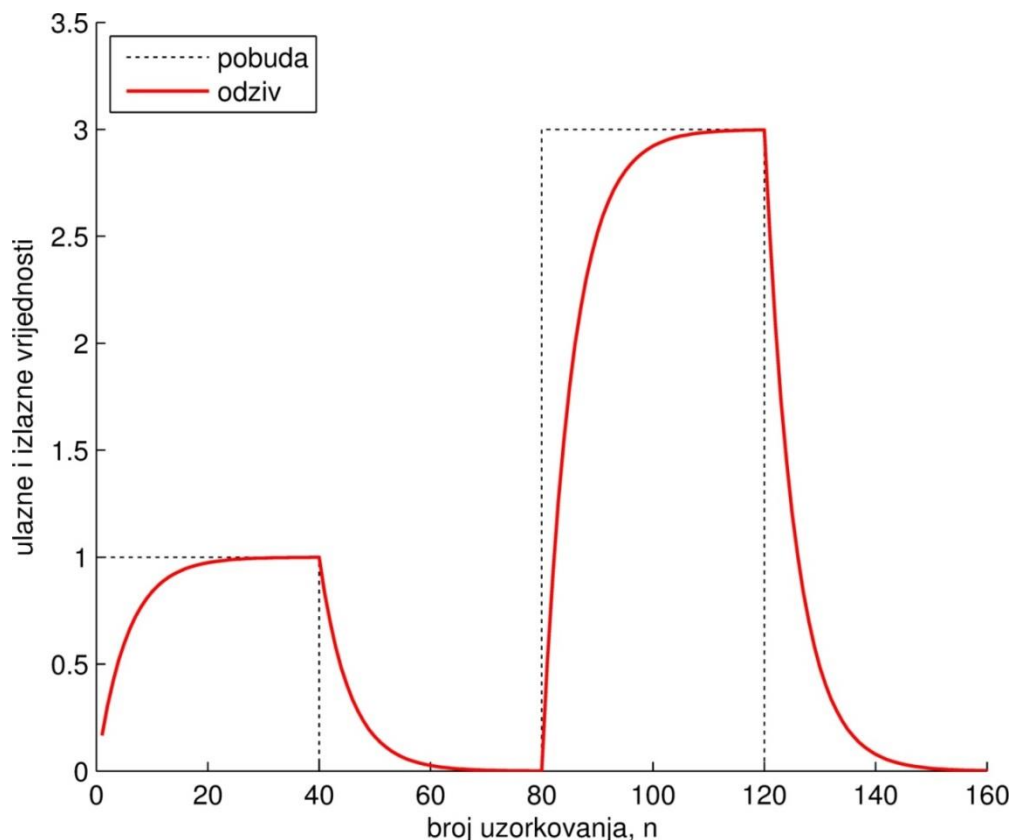
$$\sigma_{d_n} = \sqrt{\frac{1}{N} \sum_{n=1}^N (d_n - \bar{d})^2}, \quad (5-3)$$

$$NRMS = \frac{\sqrt{\frac{\sum_{n=1}^N (d_n - O_n)^2}{N}}}{\sigma_{d_n}}. \quad (5-4)$$

Prednost NRMS mjere točnosti je njena bezdimenzionalnost koja osigurava neovisnost mjere o dimenzijama učenih veličina i koja omogućuje usporedbu izvedenih algoritama učenja s drugim algoritmima, neovisno o korištenoj sklopovskoj ili programskoj podršci [3].

<sup>1</sup> Normalizirani korijen srednje kvadratne pogreške (eng. Normalized Root Mean Square error)

### 5.1. Linearni proporcionalni član prvog reda (P1)



Slika 5.3 Odziv P1 člana

Zadani primjer (Slika 5.3) koristi se kao skup za učenje mreže gdje su pobude dio ulaznog skupa podataka  $\mathbf{Z}$ , a odzivi traženog izlaznog skupa  $\mathbf{d}$ . Nakon učenja, mreža je testirana sa šest pobuda različitih intenziteta u rasponu  $[0.5, 3]$ . Mreži je potrebno normirati podatke prije učenja na raspon  $[-1, 1]$ . Budući da je najveća vrijednost 3, ulazi i izlazi skupa za učenje i skupa za testiranje dijele se s 3.

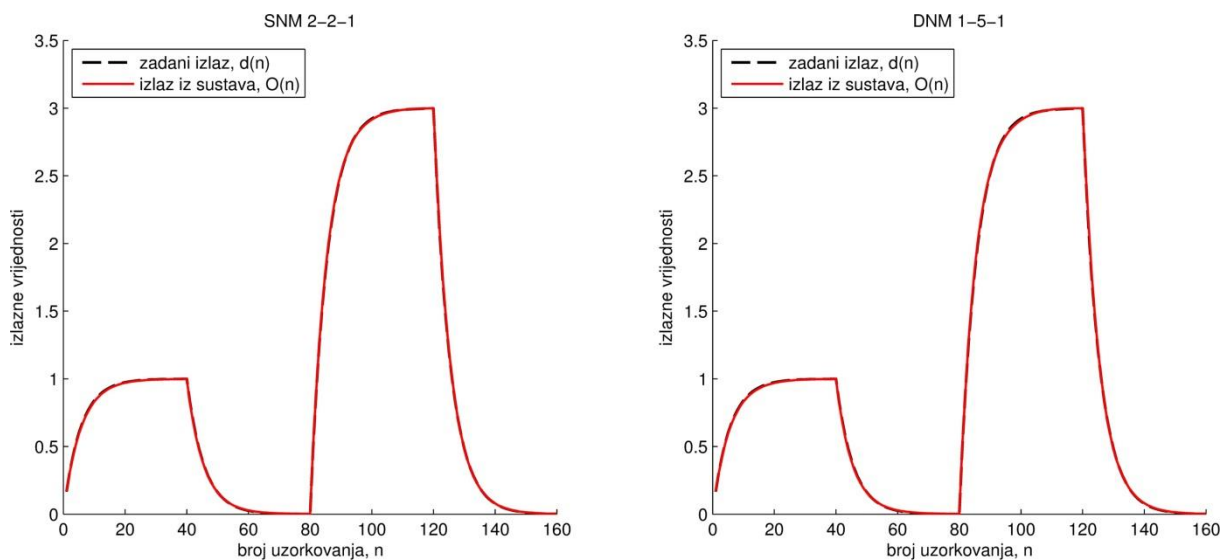
Tablica 5.1 prikazani su rezultati učenja i testiranja statičkih i dinamičkih mreža, podaci su diskretizirani prema broju neurona u skrivenom sloju. NRMS učenja predstavlja vrijednost izmjerene pogreške tokom samog postupka učenja. Ponovljeni NRMS (ili pravi NRMS) računa se nuliranjem internih parametara naučene mreže, zadržavajući jedino naučene težinske koeficijente ( $\mathbf{V}, \mathbf{W}, \mathbf{A}$  i  $\mathbf{B}$ ), te prolaženjem kroz unaprijednu fazu sa skupom vrijednosti za učenje. Prosječni NRMS je prosjek pogreške provedenih testova.



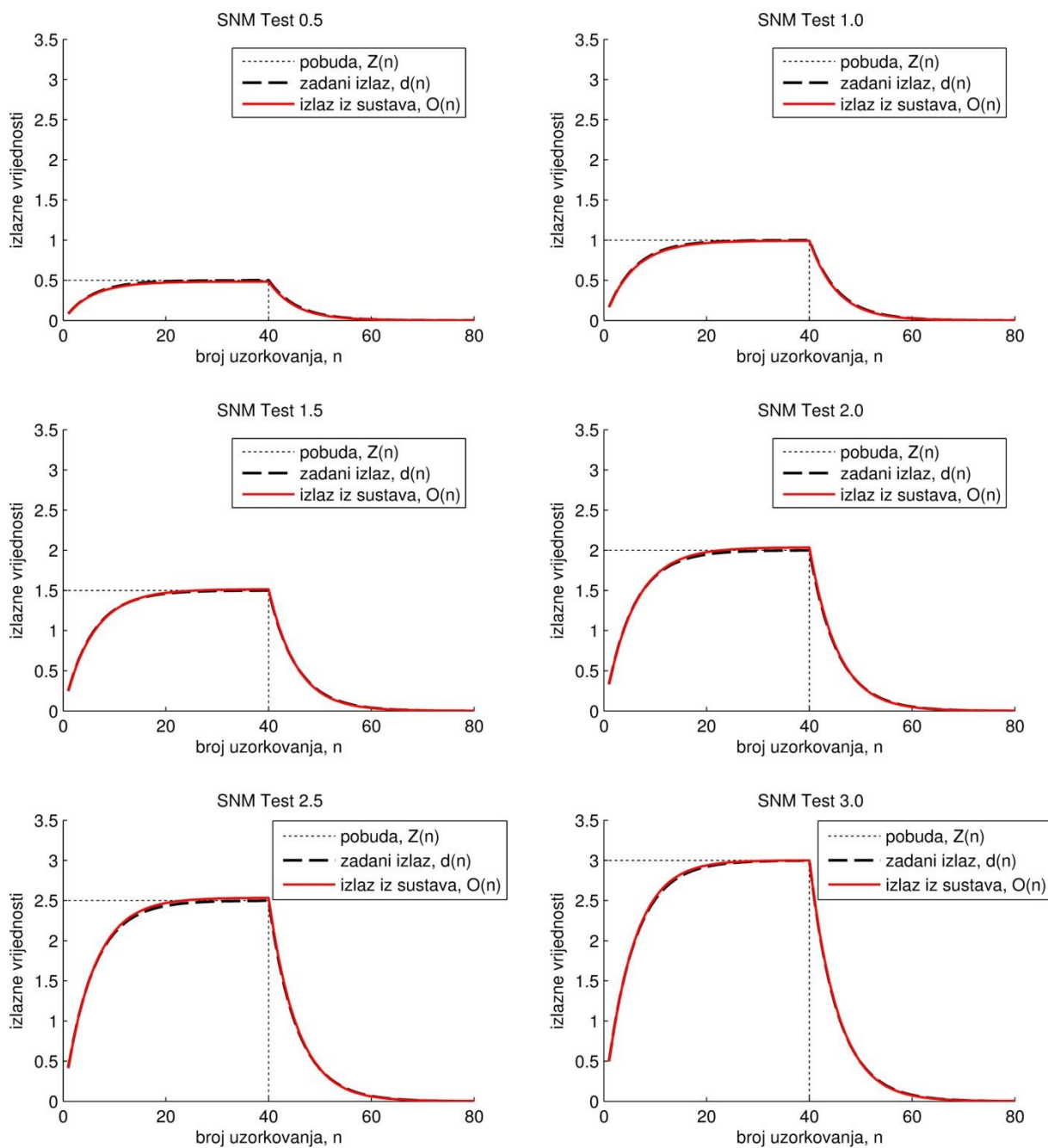
**Tablica 5.1 Usporedba algoritama pomoću NRMS mjere (linearni sustav)**

Linearni sustav	Statička mreža			Dinamička mreža		
koeficijent brzine učenja:	0,01					
momentum:	0,8					
broj iteracija:	40 000					
broj skrivenih neurona:	2	5	10	2	5	10
NRMS učenja:	0,0017	0,0013	0,0016	0,0044	0,0060	0,0039
NRMS ponovljeni:	0,0089	0,0062	0,0019	0,0056	0,0094	0,0042
NRMS test (0.5):	0,0533	0,0151	0,0155	0,0297	0,0348	0,0227
NRMS test (1.0):	0,0194	0,0080	0,0270	0,0121	0,0142	0,0094
NRMS test (1.5):	0,0130	0,0201	0,0307	0,0065	0,0080	0,0053
NRMS test (2.0):	0,0233	0,0235	0,0273	0,0088	0,0096	0,0071
NRMS test (2.5):	0,0211	0,0178	0,0178	0,0059	0,0058	0,0055
NRMS test (3.0):	0,0083	0,0068	0,0072	0,0054	0,0102	0,0039
prosječni NRMS:	0,0231	0,0152	0,0209	0,0114	0,0138	0,0090

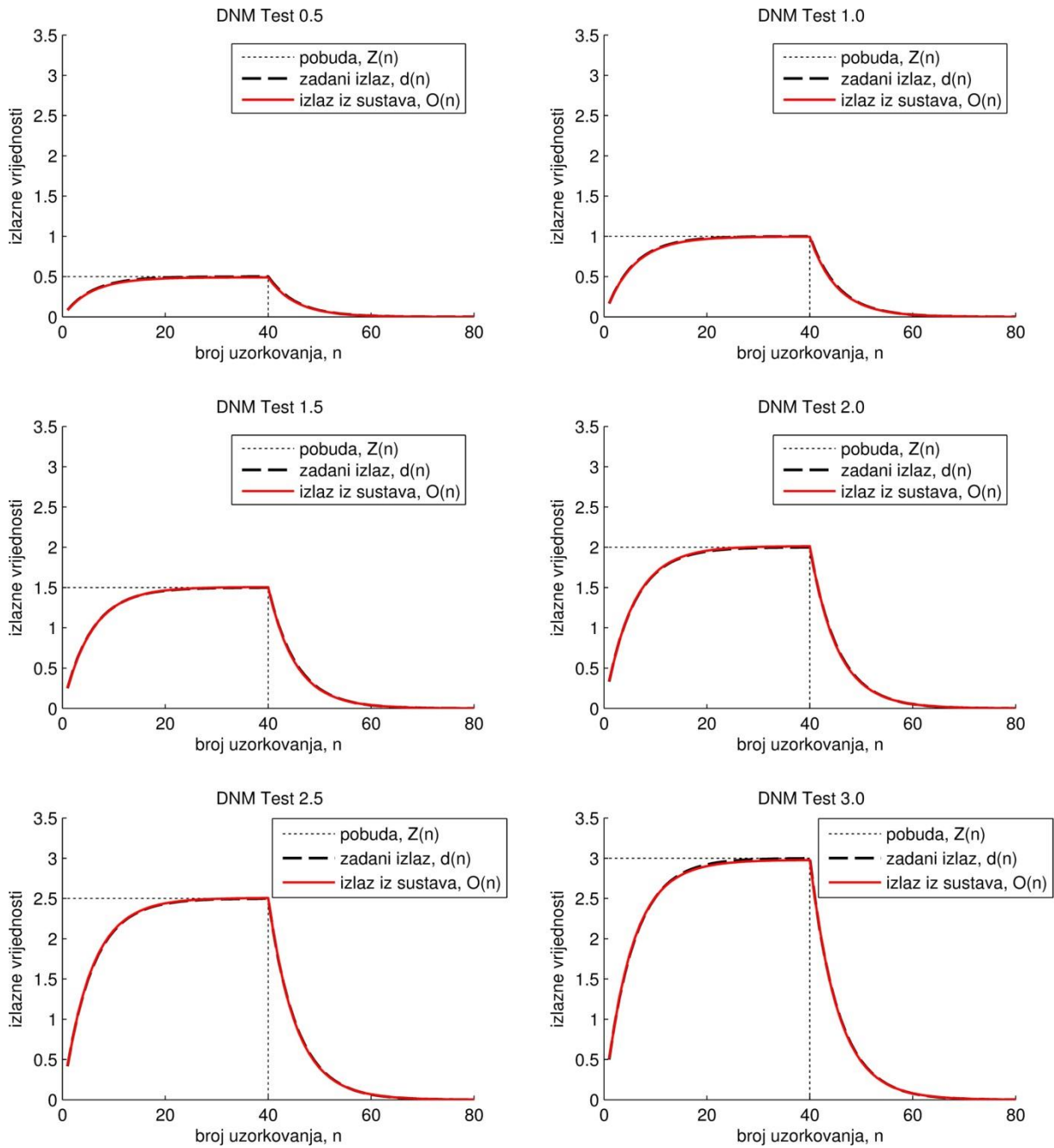
Uspoređujući NRMS mjeru učenja dvaju neuronskih mreža s najgorom prosječnom NRMS mjerom (statička mreža s 2 skrivena neurona i dinamička s 5), i iz slike njihovih odziva (Slika 5.4), vidljivo je da značajne razlike nema.

**Slika 5.4 Usporedba odziva statičke (lijevo) i dinamičke mreže (desno) pri učenju**

Slikama Slika 5.5 i Slika 5.6 uspoređuju se odzivi mreža s najvećom pogreškom na testne pobude u rasponu [0.5,3].



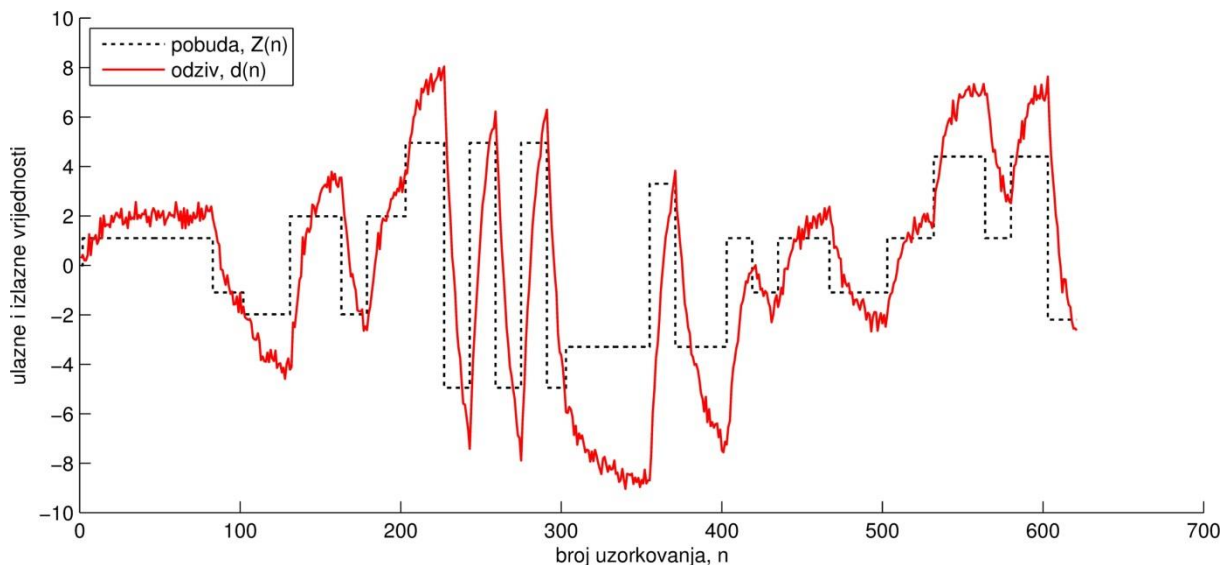
Slika 5.5 Testovi statičke mreže s 2 skrivena neurona



**Slika 5.6 Testovi dinamičke mreže s 5 skrivenih neurona**

Iz slika se može zaključiti da obje mreže vjerno oponašaju odziv P1 člana uz tek minimalnu pogrešku, međutim dinamička mreža je u većini slučajeva ipak točnija.

## 5.2. Nelinearni sustav



Slika 5.7 Odstziv nelinearnog sustava

Drugi sustav (Slika 5.7) opisan je nelinearnom jednadžbom diferencije prvog reda uz period uzorkovanja od 1 sekunde, te vremensku konstantu sustava od 10 sekundi:

$$x(n + 1) = (0.9 - 0.003x(n))x(n) + 0.2u(n) . \quad (5-5)$$

Cilj je dobiti izlazni signal koji prikazuje sve karakteristične modove procesa. Stoga se kao pobudni ulazni signal  $Z$  koristi tzv. pseudo-binarni signal slučajne frekvencije i slučajne amplitude ili APRBS<sup>2</sup>. Radi bolje simulacije realnih mjerenja, izlazni signal dodatno je opterećen šumom.

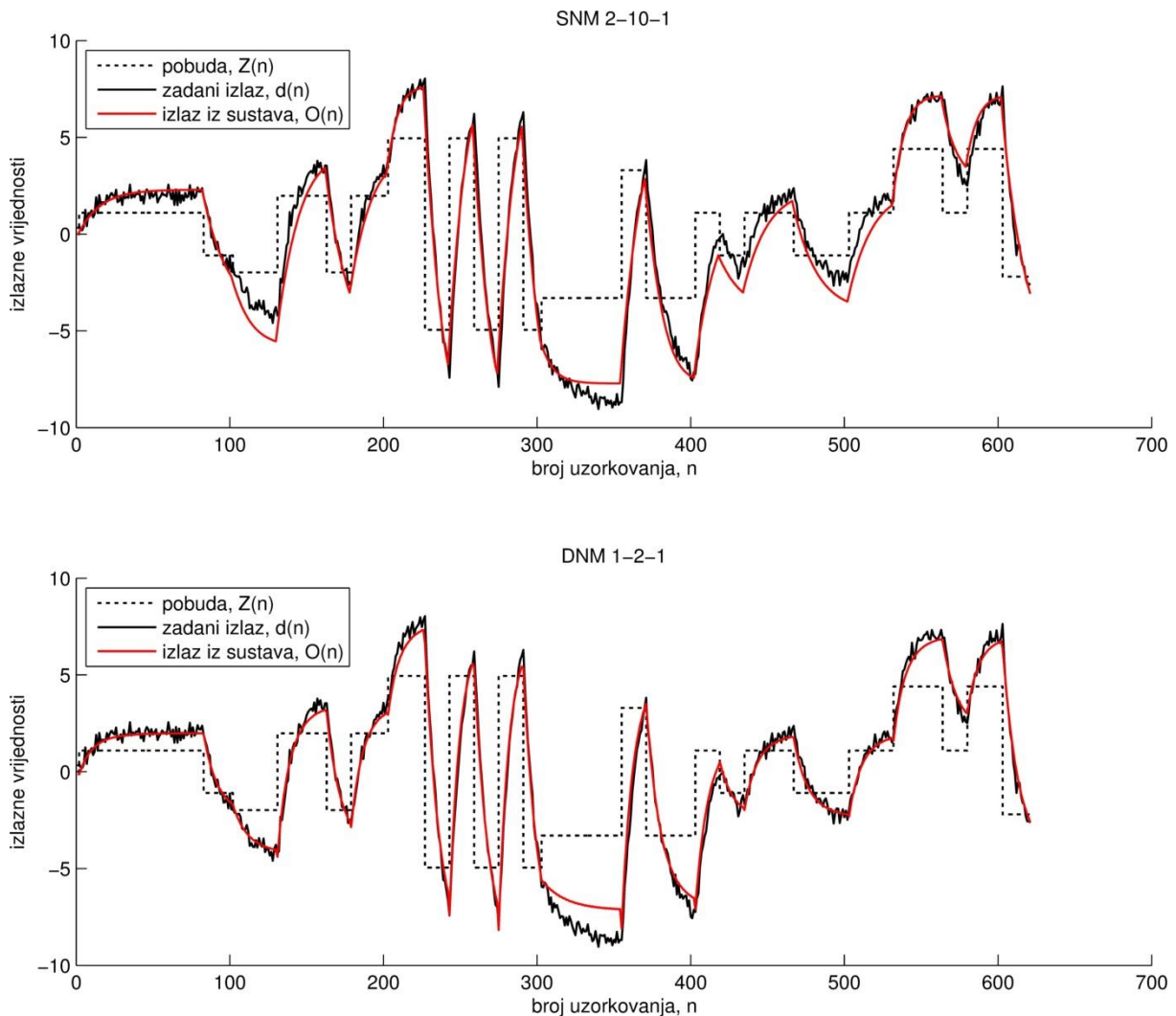
Tablicom Tablica 5.2 prikazana je usporedba uspješnosti učenja i testiranja neuronskih mreža.

Tablica 5.2 Usporedba algoritama pomoću NRMS mjere (nelinearni sustav)

Nelinearni sustav	Statička mreža			Dinamička mreža		
	2	5	10	2	5	10
koeficijent brzine učenja:	0,01					
momentum:	0,8					
broj iteracija:	15 000					
broj skrivenih neurona:	2	5	10	2	5	10
NRMS učenja:	0,1010	0,1003	0,1004	0,0989	0,0997	0,0997
NRMS ponovljeni:	0,1708	0,1765	0,1751	0,1286	0,1280	0,1170
NRMS test 1:	0,1934	0,2048	0,2102	0,0969	0,0962	0,0858
NRMS test 2:	0,2389	0,2419	0,2469	0,0742	0,0737	0,0691
NRMS test 3:	0,1977	0,1981	0,2084	0,0790	0,0784	0,0704
prosječni NRMS:	0,2100	0,2149	0,2218	0,0834	0,0828	0,0751

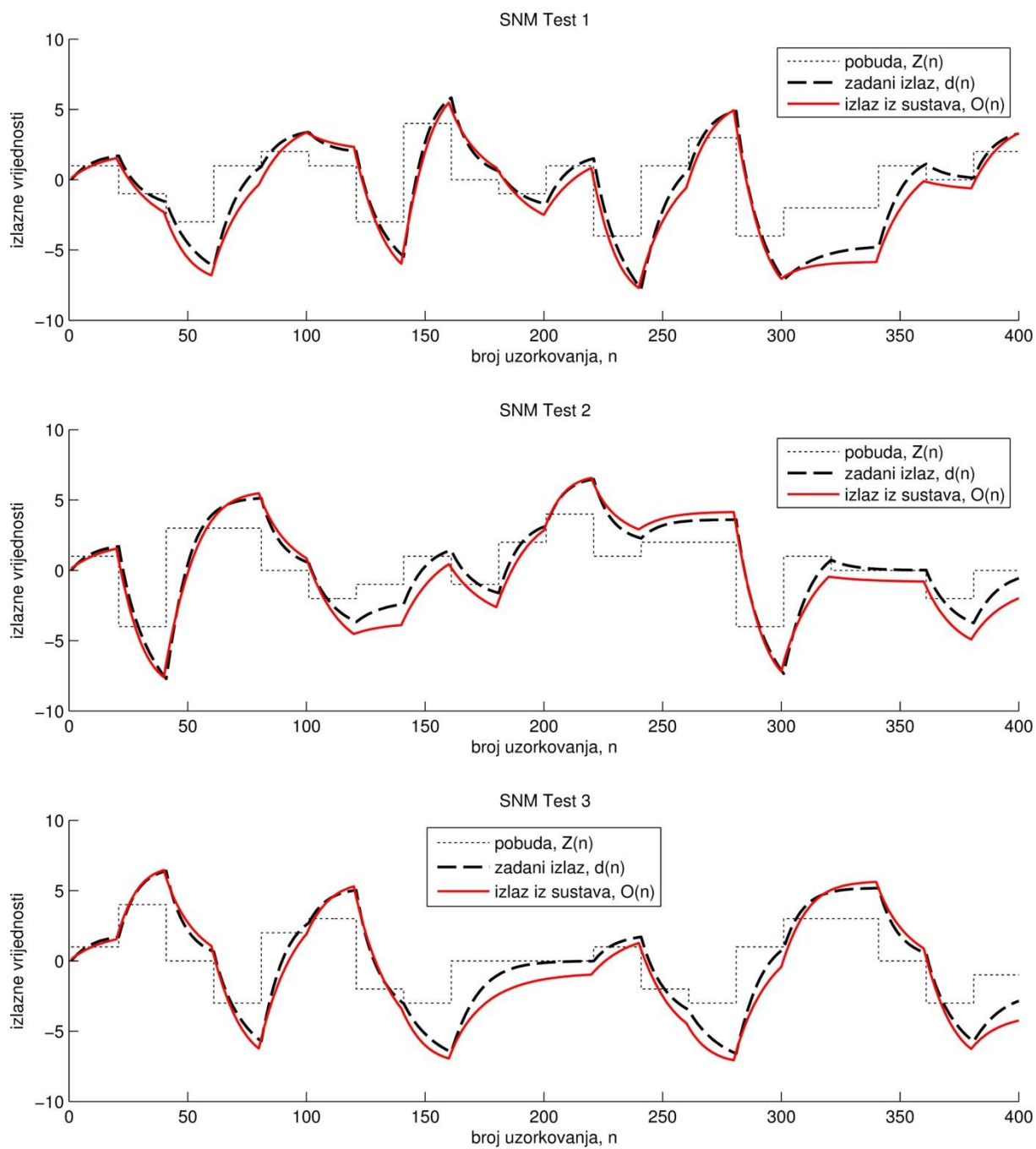
<sup>2</sup> APRBS – eng. Amplitude modulated Pseudo-Random Binary Signal

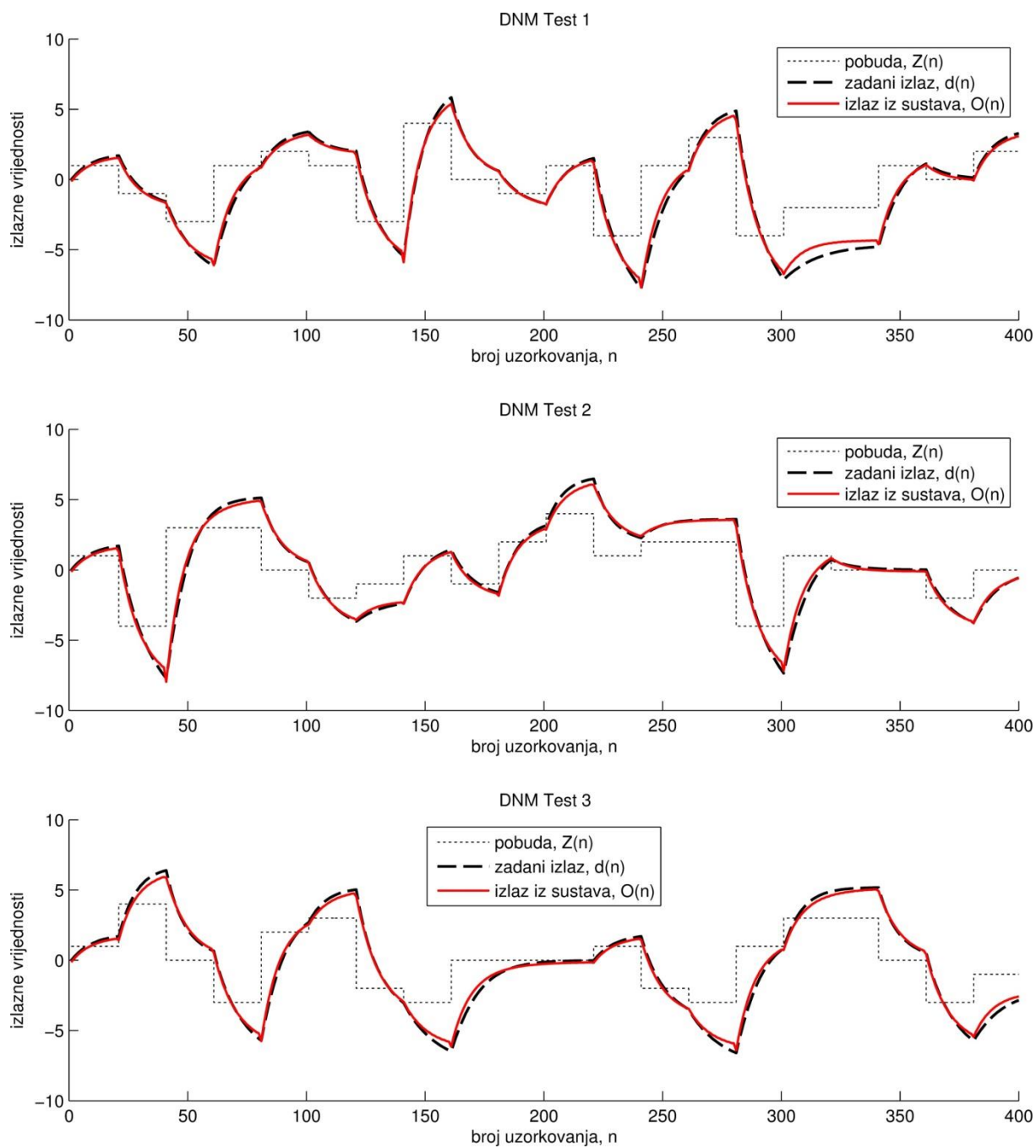
Kao i prije, prikazani su odzivi dvaju mreža s najvećom prosječnom pogreškom na podatke učenja (Slika 5.8).



**Slika 5.8** Usporedba odziva statičke (gore) i dinamičke mreže (dolje) na skup podataka učenja

Uspoređi li se ponovljena NRMS mjera pogreške statičke i dinamičke mreže Tablica 5.2, te slikom odziva najlošijih primjera (Slika 5.8), primjećuje se bolje svojstvo generalizacije dinamičkih mreža nad statičkim. A daljnjim testovima, prikazanim slikama Slika 5.9 i Slika 5.10, prednost dinamičkih mreža postaje još očitija.

**Slika 5.9 Testovi statičke mreže s 10 skrivenih neurona**



Slika 5.10 Testovi dinamičke mreže s 2 skrivena neurona

## **ZAKLJUČAK**

Izbor odgovarajuće neuronske mreže uvelike ovisi o zadatku kojeg mreža treba obaviti. Statičke neuronske mreže pokazale su se adekvatnima pri identifikaciji jednostavnih linearnih sustava, no ne i pri identifikaciji nelinearnih. Također, statičke neuronske mreže strukturno su znatno jednostavnije od dinamičkih, što pogodno utječe i na kompleksnost sklopovske realizacije. Unatoč tome, dinamičke mreže s manjim brojem umjetnih neurona, u odnosu na statičke mreže, bolje generaliziraju odziv sustava.



## LITERATURA

- [1] **James, William:** *Psychology (Briefer Course)*, H. Holt & Co, New York, 1890.
- [2] **McCulloch, Warren S i Pitts, Walter:** *A logical calculus of the ideas immanent in nervous activity*, Bulletin of Mathematical Biology, Svez. 5, 0092-8240, 1943.
- [3] **Novaković, Branko, Majetić, Dubravko i Široki, Mladen:** *Umjetne neuronske mreže*, Fakultet strojarstva i brodogradnje, Zagreb, 1998.