

**SVEUČILIŠTE U SPLITU
FAKULTET ELEKTROTEHNIKE, STROJARSTVA I
BRODOGRADNJE**

**POSLIJEDIPLOMSKI DOKTORSKI STUDIJ
ELEKTROTEHNIKE I INFORMACIJSKE TEHNOLOGIJE**

KVALIFIKACIJSKI ISPIT

**RAZVOJ I ULOGA MODELA NEURONA U
NEURONSKIM MREŽAMA**

Antonela Prnjak Gadža

Split, travnja 2026.

Sadržaj

1	Uvod.....	1
2	Umjetni neuron.....	3
2.1	Povijesni razvoj.....	3
2.2	Aktivacijske funkcije.....	4
2.3	Ograničenja klasičnih modela.....	13
3	Biološki modeli neurona.....	14
3.1	Anatomija neurona.....	14
3.2	Klasifikacija neurona.....	15
3.3	Akcijski potencijal i refraktornost.....	16
3.4	Vrijeme kao nositelj informacije i strategije kodiranja.....	18
3.4.1	Kodiranje učestalosti.....	19
3.4.2	Vremensko kodiranje.....	21
3.5	Biofizički modeli neurona.....	22
3.5.1	Leaky Integrate-and-Fire model neurona.....	22
3.5.2	Izhikevich model neurona.....	25
3.5.3	Hodgkin-Huxley model neurona.....	26
4	Arhitektura impulsnih neuronskih mreža (SNN).....	28
4.1	Topologija impulsnih neuronskih mreža.....	28
4.1.1	Koncept rijetkosti i biološka adaptacija.....	29
4.2	Učenje u SNN-ovima.....	30
4.2.1	Nadzirano učenje.....	32
4.2.2	Nenadzirano učenje.....	33
4.2.3	Učenje potkrepljivanjem.....	37
5	Implementacijski okviri i primjene SNN-a.....	39
5.1	Neuromorfno inženjerstvo i hardverske realizacije.....	39
5.2	Programski okviri za simulaciju SNN-a.....	41
6	Pregled područja.....	43
6.1	Zastupljenost modela neurona.....	44
6.2	Utjecaj modela neurona na performanse i složenost SNN sustava.....	45
6.3	Analiza sličnosti i razlika.....	48
6.4	Analiza ograničenja i nedostataka u postojećoj literaturi.....	49
7	Zaključak.....	51
	Literatura.....	52
	Popis oznaka i kratica.....	66
	Sažetak.....	67

1 Uvod

Desetljeća neuroznanstvenih istraživanja pružila su uvid u strukturu i dinamiku ljudskog mozga, čije su osnovne procesne jedinice, neuroni, povezani u složene mreže sposobne za učenje, prilagodbu i obradu informacija u stvarnom vremenu. Upravo je biološka učinkovitost mozga u rješavanju kompleksnih kognitivnih problema postala primarni izvor inspiracije za razvoj sustava umjetne inteligencije (UI). Težnja ka implementaciji biološki utemeljenih principa u računalne modele potaknula je napredak područja UI, rezultirajući razvojem modela poput konvolucijskih i rekurentnih neuronskih mreža te arhitektura temeljenih na transformerima koji postižu vrhunske rezultate u područjima računalnog vida, obrade prirodnog jezika i govora [1], [2], [3]. Navedeni modeli samo su djelomično inspirirani biološkim sustavima [4], [5], pa se njihovo funkcioniranje u praksi svodi na visokoučinkovite matematičke aproksimatore temeljene na kontinuiranim aktivacijskim funkcijama i paralelnom matričnom množenju [6]. Takav pristup zanemaruje složenu vremensku dinamiku i energetske učinkovitost koja karakterizira biološke neurone.

Ipak, unatoč iznimnim performansama, modeli neuronskih mreža suočeni su s problemom energetske neodrživosti. Treniranje i izvođenje modela zahtijeva znatne količine električne energije, što predstavlja značajan ekološki i ekonomski izazov [7], [8], [9]. Jedan od glavnih uzroka tog problema leži u Von Neumannovoj arhitekturi računala, gdje stalni prijenos podataka između procesora i memorije stvara usko grlo, povećava latenciju i nepotrebnu potrošnju energije. Dodatni ograničavajući čimbenik je usporavanje Mooreova zakona. Iako je ovaj trend desetljećima omogućavao kontinuirani rast računalne snage uz smanjenje troškova, posljednjih godina postaje sve očitije da se približava svojim fizičkim i tehnološkim granicama. U kontekstu Interneta stvari (engl. *Internet of Things*, IoT) i autonomnih perifernih sustava (*Edge AI*) sve je izraženija potreba za energetski učinkovitijom paradigmatom računanja.

Kao odgovor na navedene izazove, istraživači se ponovno okreću biološkom uzoru - ljudskom mozgu. Ljudski mozak prosječno troši oko 20 W energije, a obavlja složene kognitivne zadatke [10]. Temeljna razlika biološke i umjetne neuronske mreže leži u načinu obrade informacija. Umjetne neuronske mreže (engl. *Artificial Neuron Networks*, ANN) koriste kontinuirane numeričke vrijednosti i sinkroni prijenos informacija dok biološki inspirirane neuronske mreže informacije prenose preko diskretnih električnih impulsa (engl. *spikes*) u asinkronom vremenu. Godine 1997. predstavljene su impulsne neuronske mreže (engl. *Spiking Neuron Networks*, SNN) koje predstavljaju spoj umjetnih neuronskih mreža i biološkog načina obrade informacija i označavaju početak treće generacije neuronskih mreža [5].

Unatoč teorijskim prednostima, SNN modeli u praksi rijetko postižu bolje rezultate od klasičnih ANN modela. Jedan od ključnih izazova u postojećoj literaturi jest nedostatak standardiziranog okvira za objektivnu usporedbu ove dvije paradigme. SNN-ovi se često evaluiraju na statičkim skupovima podataka razvijenim za ANN modele, čime se ne iskorištava njihova vremenska dinamika. Kako bi se objektivno procijenile prednosti i ograničenja SNN-ova u odnosu na klasične neuronske mreže, nužno je proširiti kriterije evaluacije izvan same točnosti. To uključuje metrike poput energetske potrošnje po operaciji, latencija donošenja odluke, vremena konvergencije te robusnosti na šum, uz korištenje skupova podataka temeljenih na događajima [11], [12].

Ovaj rad istražuje razlike između *klasičnih* i biološki vjernijih umjetnih neurona, s posebnim naglaskom na SNN-ove. Cilj je sustavno prikazati povijesni i konceptualni razvoj neuronskih mreža, analizirati prednosti i ograničenja ANN-ova i SNN-ova, te ukazati na potencijalne

smjerove budućeg razvoja i primjene, osobito u kontekstu energetski učinkovitog neuromorfnog računarstva¹.

Rad je strukturiran na sljedeći način. Drugo poglavlje uvodi temeljne koncepte umjetnog neurona, uključujući formalni matematički model i aktivacijske funkcije, dok se treće poglavlje fokusira se na biološke neurone i na biofizičke modele koji nastoje preciznije opisati složene električne i vremenske procese u istima. Četvrto poglavlje posvećeno je arhitekturi SNN-ova s naglaskom na način prijenosa informacija putem diskretnih impulsa i metode učenja. Peto poglavlje donosi hardverske realizacije i programske okvire za simulaciju SNN-ova. Posljednje poglavlje je pregled područja.

¹ Neuromorfno računarstvo predstavlja način razvoja računala koja strukturno i funkcionalno oponaša principe rada ljudskog mozga.

2 Umjetni neuron

Neuroni su temeljni gradivni element neuronskih mreža te izravno određuju način obrade, prijenosa i kodiranja informacija unutar sustava. Razlike u formulaciji neuronskih modela ne očituju se samo u njihovom matematičkom opisu, već imaju izravan učinak na arhitekturu mreže, mehanizme učenja, vremensku dinamiku obrade i energetske učinkovitost.

U području umjetne inteligencije dominiraju dva pristupa modeliranju. Prvi se temelji na apstraktnim, matematički optimiziranim neuronima koji koriste kontinuirane aktivacijske funkcije i sinkroni prijenos signala. Takvi modeli omogućuju učinkovitu primjenu gradijentnih metoda optimizacije, ali zanemaruju ključne dinamičke značajke bioloških sustava. Drugi pristup nastoji vjernije replicirati biološke mehanizme komunikacije uvođenjem vremenske dimenzije i asinkrone, događajima vođene obrade informacija.

U okviru prvog pristupa, *klasični* umjetni neuron predstavlja visoko apstrahiran matematički model biološkog neurona čija je uloga nelinearna transformacija ulaznih signala u izlazne vrijednosti. Unatoč biološkoj inspiraciji, ovaj model zanemaruje većinu strukturnih i dinamičkih karakteristika živčanih stanica, zadržavajući formalno jednostavan i računski pogodan opis.

2.1 Povijesni razvoj

Povijesni razvoj započinje McCulloch-Pittsovim modelom iz 1943. godine, koji predstavlja prvi formalizirani pokušaj modeliranja neurona sposobnog za izvođenje logičkih operacija i procesa donošenja odluka [13]. Model sažima ideju prema kojoj neuroni primaju, integriraju i prenose informacije. Model se temelji na nizu binarnih ulaza (x_1, x_2, \dots, x_n) i pripadajućih težina (w_1, w_2, \dots, w_n) koje predstavljaju intenzitet utjecaja pojedinog ulaza. Neuron izračunava ponderiranu sumu ulaza i uspoređuje ju s unaprijed definiranim pragom θ . Ako suma dosegne ili premaši prag, neuron generira binarni izlaz (1), u protivnom miruje (0). Taj mehanizam predstavlja začetni oblik aktivacijske funkcije. Formalno:

$$y = \begin{cases} 1, & \text{ako } \sum_{i=1}^n w_i x_i \geq \theta \\ 0, & \text{inače} \end{cases} \quad (2.1)$$

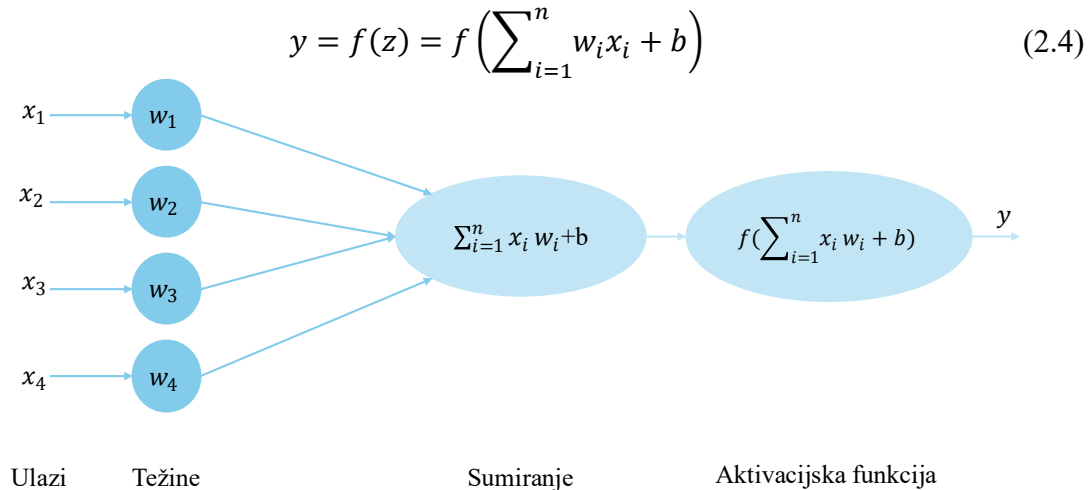
Značajan iskorak učinio je Rosenblattov perceptron uvođenjem algoritma učenja koji omogućuje automatsko podešavanje težina na temelju pogrešaka u predviđanju. Za razliku od prethodnih statičnih modela, perceptron uvodi i parametar pomaka b (engl. *bias*). Pomak služi kao dodatni „uteg“ za fino podešavanje praga aktivacije i omogućuje neuronu fleksibilnije pozicioniranje granice odlučivanja [14]. Model se definira kao:

$$y = \begin{cases} 1, & \text{ako } \sum_{i=1}^n w_i x_i + b \geq \theta \\ 0, & \text{inače} \end{cases} \quad (2.2)$$

U općenitom obliku, obrada informacija u klasičnom umjetnom neuronu odvija se u dva koraka. U prvom se koraku izračunava ponderirana suma ulaznih signala; neuron prima ulazne vrijednosti x_i , množi ih s pripadajućim težinama w_i te im pribraja pomak b , čime se dobiva neto ulaz z , linearna kombinacija ulaznih podataka:

$$z = \sum_{i=1}^n w_i x_i + b \quad (2.3)$$

Dobiveni rezultat je još uvijek samo jednostavna linearna kombinacija brojeva. Kako bi neuron mogao rješavati složene probleme (koji u stvarnom svijetu rijetko slijede ravne linije), neto ulaz se proslijeđuje aktivacijskoj funkciji $f(\cdot)$ čija je uloga transformirati linearni ulaz u nelinearan izlazni signal y (Slika 2.1). Na taj način neuronska mreža može prepoznati složenije uzorke i odnose u podacima.



Slika 2.1 Matematički model umjetnog neurona

2.2 Aktivacijske funkcije

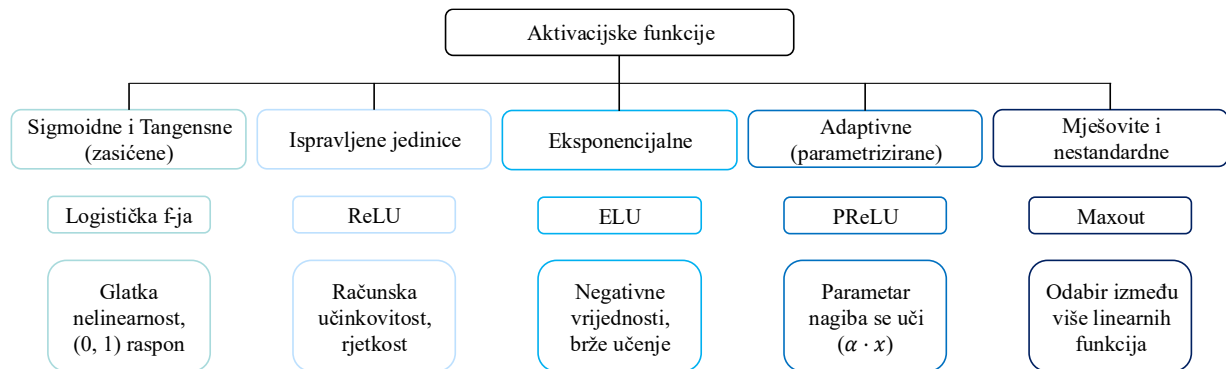
Aktivacijske funkcije određuju intenzitet signala koji se prenosi na sljedeći sloj, djelujući kao filter koji odlučuje o važnosti informacije a bez kojih bi mreža, bez obzira na dubinu, bila ekvivalentna jednoslojnom linearnom modelu, nesposobna za rješavanje linearno nerazdvojivih problema. Čak i temeljni logički problem poput operacije isključivo-ili (XOR) zahtijeva nelinearnost, a složenost stvarnih podataka poput slike, govora i medicinskih zapisa to dodatno naglašavaju. Teorijsku potvrdu nelinearnosti daje teorem univerzalne aproksimacije [15], [16]. Pokazano je da jednoslojna neuronska mreža s kontinuiranom, ograničenom i nekonstantnom aktivacijskom funkcijom sigmoidnog tipa može aproksimirati svaku kontinuiranu funkciju na kompaktnim skupovima [15]. Kasnije je pokazano da univerzalna aproksimacija vrijedi za široku klasu nelinearnih aktivacijskih funkcija, bez zahtjeva za ograničenošću ili monotonijom [16]. Ta svojstva omogućuju modelu oblikovanje složenih granica odlučivanja i učenje apstraktnih reprezentacija podataka unutar višedimenzionalnog prostora značajki.

U literaturi aktivacijske funkcije sistematiziraju se kroz različite taksonomije, ovisno o njihovoj matematičkoj strukturi i biološkoj inspiraciji. Jedna od čestih podjela razlikuje dvije glavne klase: fiksne i adaptivne funkcije [17]. Fiksne (klasične) aktivacijske funkcije unaprijed su definirani matematički izrazi koji primjenjuju istu transformaciju na sve ulaze, neovisno o njihovim vrijednostima. Tipični primjeri uključuju logističku sigmoidnu funkciju [18], hiperbolni tangens [19] i ReLU [20]. S druge strane, adaptivne aktivacijske funkcije uče svoje parametre na temelju ulaznih podataka, što im omogućuje promjenu oblika tijekom treniranja. Primjeri takvih funkcija su PReLU [21] i PELU [22].

Druga klasifikacija se naslanja na prvu pa razlikuje funkcije fiksnog oblika (klasične i ReLU-temeljene funkcije) i funkcije koje se mogu učiti, a potonje se dodatno se dijele na parametrizirane standardne funkcije i funkcije temeljene na metodama ansambla [23]. Posebnu kategoriju čine nestandardne funkcije gdje nelinearnost nije zadana jednom skalarnom funkcijom nad linearnom kombinacijom ulaza, već se ostvaruje kroz strukturalno definirano natjecanje između više linearnih odgovora (npr. Maxout [24] i LWTA[25]).

Treća klasifikacija grupira aktivacijske funkcije na fiksne, adaptivne i moderne funkcije (uključujući parametarske i ansambl-aktivacije) te nestandardne funkcije koje koriste izlaze prethodnih slojeva [26].

Izbor taksonomije u ovom radu motiviran je analitičkom pogodnošću. Korištena podjela razlikuje pet skupina: sigmoidne i tangensne funkcije, ispravljene jedinice, eksponencijalne funkcije, adaptivne funkcije te mješovite funkcije namijenjene optimizaciji performansi u specijaliziranim arhitekturama [27] (Slika 2.2). Ova podjela obuhvaća temeljne matematičke mehanizme uvođenja nelinearnosti, čime omogućuje sustavnu usporedbu funkcija unutar i između skupina, dok broj skupina ostaje dovoljno ograničen da spriječi pretjeranu rascjepkanost prostora aktivacijskih funkcija.



Slika 2.2 Taksonomija aktivacijskih funkcija prema [27]. Prikazana je hijerarhijska podjela na pet ključnih skupina s pripadajućim reprezentativnim primjerima.

Valja naglasiti da granice između navedenih skupina u praksi nisu stroge. Mnoge aktivacijske funkcije mogu se promatrati kao varijacije ili blago izmijenjeni oblici nekoliko osnovnih funkcija. Razlike unutar iste skupine najčešće su kvantitativne, primjerice u nagibu ili glatkoći, dok osnovni oblik funkcije ostaje isti [28]. Zbog toga je izbor između sličnih funkcija češće uvjetovan njihovim numeričkim svojstvima nego razlikama u načinu na koji su definirane.

U nastavku će se analiza temeljiti na usporednom prikazu reprezentativnih primjera iz svake skupine, s naglaskom na njihove sličnosti. Ovakav pristup omogućuje jasnije razumijevanje u kojoj mjeri različite aktivacijske funkcije uvode novu mogućnost modeliranja, a u kojoj mjeri predstavljaju samo prilagodbe već postojećih oblika nelinearnosti.

Aktivacije funkcije temeljene na sigmoidnim i tangens funkcijama

U ranim fazama razvoja neuronskih mreža, logistička sigmoidna funkcija i hiperbolni tangens bile su primarni izbor za uvođenje nelinearnosti, jer obje funkcije nastoje oponašati način aktivacije bioloških neurona kroz nelinearno zasićenje.

Logistička sigmoidna funkcija predstavlja povijesno najpopularniju nelinearnu transformaciju u umjetnim neuronskim mrežama. Definirana je izrazom:

$$f(x) = \frac{1}{1 + e^{-x}} \quad (2.5)$$

Funkcija preslikava izlazne vrijednosti u strogo definirani raspon $[0, 1]$, glatka je, strogo monotona i diferencijabilna na cijeloj domeni. Unatoč tim pogodnim svojstvima, u dubokim arhitekturama pokazuje dva ključna nedostatka. Prvo, u područjima zasićenja, derivacija poprima male vrijednosti pa se gradijent (veličina koja određuje koliko se model mijenja tijekom učenja) pri propagaciji kroz slojeve postupno prigušuje, pojava poznata kao problem

išchezavajućeg gradijenta [29]. Drugo, asimetrija izlaznog raspona uzrokuje pomak srednje vrijednosti aktivacija, što može usporiti konvergenciju [19].

Funkcija hiperbolnog tangensa uvedena je kao poboljšanje sigmoidne, prvenstveno zbog svojstva centriranosti oko nule. Definirana je kao:

$$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (2.6)$$

Funkcija preslikava ulaz u interval $[-1, 1]$ i zadržava S-oblik sigmoidne funkcije, pri čemu je razlika između dvije opisane funkcije kvantitativna: hiperbolni tangens je pomak i skaliranje istog funkcionalnog obrasca. Centriranost oko nule ublažava problem neuravnoteženih gradijenata i u praksi često ubrzava konvergenciju. Međutim, hiperbolni tangens dijeli temeljno ograničenje sigmoidne funkcije - izražen problem išchezavajućeg gradijenta u područjima zasićenja i relativno visoku računalnu složenost zbog eksponencijalnih operacija [30].

Strukturna homogenost obitelji

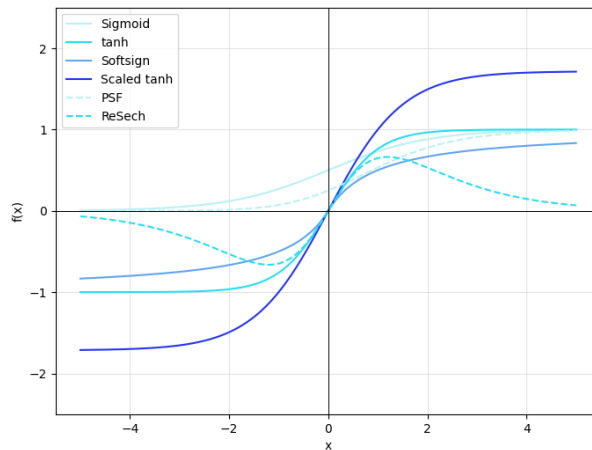
Ostale funkcije unutar ove obitelji mogu se interpretirati kao deformacije osnovnog prototipa kroz promjenu nagiba, amplitudnog raspona ili uvođenje dodatnog parametra za kontrolu strmine. Sve zadržavaju tri ključne značajke: glatkoću i diferencijabilnost, zasićenje pri velikim vrijednostima i maksimalnu derivaciju u blizini ishodišta. Razlike su stoga primarno kvantitativne, dok je kvalitativni mehanizam nelinearnosti identičan (Tablica 2.1).

Tablica 2.1 Sažetak aktivacijskih funkcija temeljenih na sigmoidnim i tangens funkcijama.

Naziv	Kratica	Godina	Formula	Raspon izlaza
Logistička sigmoidna funkcija	-	-	$\frac{1}{1 + e^{-x}}$	$[0, 1]$
Hiperbolni tangens	Tanh	-	$\frac{e^x - e^{-x}}{e^x + e^{-x}}$	$[-1, 1]$
SoftSign	-	-	$\frac{x}{1 + x }$	$[-1, 1]$
Skalirani hiperbolni tangens [31]	sTanh	1998	$A \times \text{Tanh}(B \times x)$	$[-A, A]$
Parametrijska sigmoidna funkcija [32]	PSF	2004	$\frac{1}{(1 + e^{-x})^m}$	$[0, 1]$
Ispravljani hiperbolni sekans [33]	ReSech	2016	$x \times \text{Sech}(x)$	$[-1, 1]$
Skalirani sigmoid [34]	sSigmoid	2016	$4 \times \text{Sigmoid}(x) - 2$	$[-2, 2]$
Penalizirani hiperbolni tangens [34]	pTanh	2016	$\begin{cases} \text{Tanh}(x), & x \geq 0 \\ a \times \text{Tanh}(x), & x < 0 \end{cases}$	$[-a, 1]$
Hexpo [35]	-	2017	$\begin{cases} -a \times (e^{-\frac{x}{b}} - 1), & x \geq 0 \\ c - \times (e^{\frac{x}{d}} - 1), & x < 0 \end{cases}$	$[-c, a]$
Poboljšani sigmoid [36]	ISigmoid	2018	$\begin{cases} \alpha \times (x - a) + \text{Sigmoid}(a), & x \geq a \\ \text{Sigmoid}(x), & -a < x < a \\ \alpha \times (x + a) + \text{Sigmoid}(a), & x \leq -a \end{cases}$	$(-\infty, \infty)$
Sigmoidno ponderirane linearne jedinice [37]	SiLU	2018	$x \times \text{Sigmoid}(x)$	$(-0.5, \infty)$
Linearno skalirani hiperbolni tangens [38]	LiSHT	2019	$x \times \text{Tanh}(x)$	$[0, \infty)$

Elliott [39]	-	2019	$\frac{0.5 \times x}{1 + x } + 0.5$	$[0, 1]$
Znak mekog korijena [40]	SRS	2020	$\frac{x}{\frac{x}{\alpha} + e^{-\frac{x}{\beta}}}$	$[\frac{\alpha \times \beta}{\beta - \alpha \times e}, \alpha]$
Parametrijske tangente curenja [41]	PLTanh	2023	$\begin{cases} \tanh(x), & \tanh(x) \geq \alpha x \\ \alpha x, & \tanh(x) \leq \alpha x \end{cases}$	$[0, \infty)$
Sigmoid-SoftSign [42]	S3	2025	$\begin{cases} \frac{1}{1 + e^{-x}}, & x \leq 0 \\ \frac{1}{1 + x }, & x > 0 \end{cases}$	$(0, 1)$

Promatrane kao cjelina, sigmoidne i tangensne funkcije čine homogenu obitelj glatkih nelinearnih funkcija sa zasićenjem saturirajućih nelinearnosti. Iz te perspektive, izbor između logističke sigmoidne funkcije, hiperbolnog tangensa ili njihovih varijanti nije izbor između različitih tipova nelinearnosti, već izbor između različitih numeričkih modifikacija istog funkcionalnog oblika. Upravo je to svojstvo zasićenja potaknulo prijelaz prema dijelom linearnim funkcijama² tipa ReLU, koje napuštaju ograničenje u pozitivnoj domeni.



Slika 2.3 Sigmoidne i tangens aktivacijske funkcije.

Sigmoidna funkcija se koristi u izlaznom sloju za binarnu klasifikaciju jer se izlaz prirodno interpretira kao vjerojatnost. Hiperbolni tangens može se koristiti u skrivenim slojevima plitkih mreža, osobito kada su podaci centrirani oko nule. U dubokim mrežama obje se funkcije zamjenjuju ReLU-om zbog problema zasićenja [43].

Ispravljene aktivacijske funkcije

Ispravljene aktivacijske funkcije predstavljaju skupinu nelinearnih funkcija koje su omogućile razvoj dubokih neuronskih mreža [20]. Najpoznatiji predstavnik je ispravljena linearna jedinica (engl. *Rectified Linear Unit*, ReLU), definirana kao:

$$f(x) = \max(0, x) = \begin{cases} x, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (2.7)$$

² Dijelom linearna funkcija je funkcija koja je linearna na odvojenim intervalima svoje domene, pri čemu linearni segmenti mogu imati različite nagibe i presjeke.

ReLU zadržava linearnost u pozitivnoj domeni, a negativne vrijednosti preslikava u nulu. Derivacija iznosi 1 za pozitivne ulaze, 0 za negativne, a u točki $x=0$ nije jednoznačno definirana, no u praksi se uzima jedna od tih vrijednosti.

U odnosu na sigmoidne funkcije, ReLU uvodi tri ključne promjene: eliminira zasićenje u pozitivnoj domeni, zadržava konstantan gradijent za aktivne neurone i reducira računalnu složenost uklanjanjem eksponencijalnih operacija. Odsutnost zasićenja u pozitivnom području omogućila je stabilniji prijenos gradijenta kroz duboke slojeve.

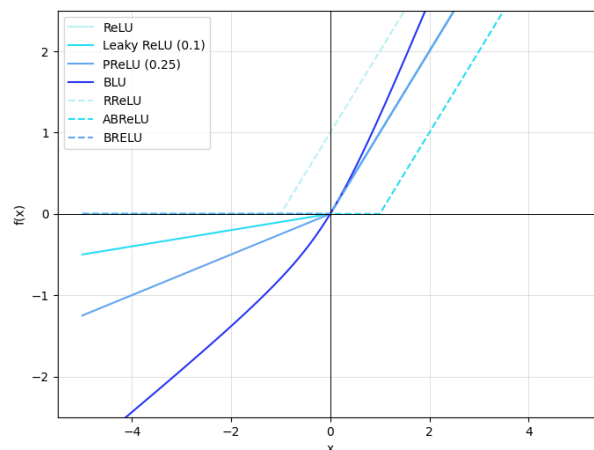
Temeljno ograničenje ReLU funkcije proizlazi iz potpunog poništavanja negativne domene. Ako neuron tijekom treniranja uđe u područje trajno negativnih ulaza, njegov izlaz ostaje nula i gradijent se prestaje propagirati, fenomen poznat kao problem „mrtvih neurona“.

Strukturna homogenost obitelji

Varijante ReLU se mogu promatrati kao prilagodbe osnovnog oblika funkcije. Propuštajuća ReLU (engl. *Leaky ReLU*, LReLU), uvodi mali, nenulti nagib $\alpha \ll 1$ u negativnom području, čime se zadržavaju informacije iz negativnih vrijednosti ulaza [44]:

$$f(x) = \begin{cases} x, & x \geq 0 \\ \alpha x, & x < 0 \end{cases} \quad (2.8)$$

Parametrom α se ublažava diskontinuitet u derivaciji između domena, dok temeljni mehanizam ostaje isti: djelomična linearnost, nepostojanje ograničenja izlaza i dominantna linearnost u aktivnoj domeni.



Slika 2.4 Ispravljene aktivacijske funkcije.

ReLU-temeljene funkcije čine homogenu obitelj dijelom linearnih funkcija s neograničenom izlaznom vrijednosti (Slika 2.4). Njihova reprezentacijska sposobnost proizlazi iz selektivne aktivacije i zadržavanja linearnog gradijenta u pozitivnoj domeni, dok ograničenja dolaze iz redukcije informacija u negativnom području. Izbor između ReLU, *Leaky ReLU* ili parametriziranih varijanti predstavlja prilagodbu intenziteta ispravljanja i kontinuiteta gradijenta, a ne promjenu vrste nelinearnosti. Pregled svih varijanti s formulama i rasponima prikazan je u Tablica 2.2.

Tablica 2.2 Sažetak ispravljenih aktivacijskih funkcija.

Naziv	Kratica	Godina	Formula	Raspon izlaza
Ispravljena linearna jedinica [20]	ReLU	2010	$\begin{cases} x, & x \geq 0 \\ 0, & x < 0 \end{cases}$	$[0, \infty)$

Propuštajuća ReLU [44]	LReLU	2013	$\begin{cases} x, & x \geq 0 \\ \alpha \times x, & x < 0 \end{cases}$	$(-\infty, \infty)$
Parametrijski ReLU [45]	PReLU	2015	$\begin{cases} x, & x \geq 0 \\ p \times x, & x < 0 \end{cases}$	$(-\infty, \infty)$
Randomizirana ReLU [46]	RReLU	2015	$\begin{cases} x, & x \geq 0 \\ R \times x, & x < 0 \end{cases}$	$(-\infty, \infty)$
Spojjeni ReLU [47]	CReLU	2016	$[ReLU(x), ReLU(-x)]$	$[0, \infty)$
Ograničeni ReLU [48]	BReLU	2016	$\min(\max(0, x), A)$	$[0, A]$
Parametrijska tanh-linearna jedinica [49]	PTELU	2017	$\begin{cases} x, & x \geq 0 \\ \alpha \times \tanh(\beta \times x), & x < 0 \end{cases}$	$[-\alpha, \infty)$
Fleksibilni ReLU [50]	FReLU	2018	$ReLU(x) + b$	$[b, \infty)$
Elastični ReLU [51]	ERReLU	2018	$\max(R \times x, 0)$	$[0, \infty)$
Nasumično translacijski ReLU [52]	RTRReLU	2018	$\begin{cases} x + a, & x + a > 0 \\ 0, & x + a \leq 0 \end{cases}$	$[0, \infty)$
Dualni ReLU [53]	DReLU	2018	$\max(0, a) - \max(0, b)$	$(-\infty, \infty)$
Upareni ReLU [54]	-	2018	$[\max(s \times x - \theta, 0), \max(s_p \times x - \theta_p, 0)]$	$(-\infty, \infty)$
Prosječno pristrani ReLU [55]	ABReLU	2018	$\begin{cases} x - \beta, & x - \beta \geq 0 \\ 0, & x - \beta < 0 \end{cases}$	$[0, \infty)$
V-oblikovani ReLU [56]	vReLU	2018	$\begin{cases} x, & x \geq 0 \\ -x, & x < 0 \end{cases}$	$[0, \infty]$
Linearna jedinica po dijelovima [57]	PLU	2018	$\max(\alpha \times (x + c) - c, \min(\alpha \times (x - c) + c, x))$	$[-\infty, \infty]$
Pomaknuti ReLU [58]	DisReLU	2019	$\begin{cases} x, & x \geq -\delta \\ -\delta, & x < -\delta \end{cases}$	$[-\delta, \infty]$
Savijljiva linearna jedinica [59]	BLU	2019	$\beta \times (\sqrt{x^2 + 1} - 1) + x$	$(-\infty, \infty)$
Prirodni logaritam [60]	NLReLU	2019	$\ln(\beta \times \max(0, x) + 1.0)$	$[0, \infty)$
Višekanalne linearne jedinice za obučavanje [61]	MTLU	2019	$\begin{cases} a_0 \times x + b_0, & x \leq c_0 \\ a_k \times x + b_k, & c_{k-1} < x \leq c_k \\ \dots \\ a_K \times x + b_K, & c_{K-1} < x \end{cases}$	$(-\infty, \infty)$
Lipschitzov ReLU [62]	L-ReLU	2020	$\begin{cases} \max(\phi(x), 0), & x \geq 0 \\ \min(\eta(x), 0), & x < 0 \end{cases}$	$(-\infty, \infty)$
Zorro [63]	-	2024	$\begin{cases} kxGS(x, a, b), & x < 0 \\ x, & 0 \leq x \leq 1 \\ 1 - k(1 - x)GS(1 - x, a, b), & x > 1 \end{cases}$ $k=1+e^{ab}$	parametarski omeđen $\approx [0, 1]$

Zbog stabilnijeg gradijenta i računalne učinkovitosti, ReLU i njezine varijante dominantne su u skrivenim slojevima dubokih konvolucijskih i potpuno povezanih mreža. *Leaky ReLU* preporučuje se kada standardni ReLU dovodi do velikog broja neaktivnih neurona, ali izbor između varijanti najčešće se temelji na empirijskom testiranju [43].

Eksponecijalne aktivacijske funkcije

Eksponecijalne aktivacijske funkcije predstavljaju prijelaznu skupinu između dijelom linearnih i glatkih funkcija s ograničenim izlazom. Razvijene su s ciljem zadržavanja dobrih karakteristika ReLU funkcije u pozitivnoj domeni uz istodobno ublažavanje problema potpunog poništavanja negativnih aktivacija. Strukturno gledano, ove funkcije kombiniraju linearnost za pozitivne ulaze s ekspancijalnom transformacijom u negativnom području.

Najpoznatija funkcija iz ove obitelji je ekspancijalna linearna jedinica (engl. *Exponential Linear Unit*, ELU), definirana kao:

$$f(x) = \begin{cases} x, & x > 0 \\ \alpha(e^x - 1), & x \leq 0 \end{cases} \quad (2.9)$$

gdje je parametar $\alpha > 0$. ELU zadržava linearnost u pozitivnoj domeni, čime omogućuje širenje gradijenta, dok u negativnom području uvodi glatku eksponencijalnu transformaciju koja asimptotski teži u $-\alpha$. Za razliku od ReLU, diferencijabilna je u cijeloj domeni, uključujući i točku prijelaza.

U odnosu na prethodne obitelji funkcija, ove funkcije nemaju zasićenje u pozitivnom području, dok se negativne vrijednosti postupno smanjuju prema donjoj granici. Funkcija je glatka i diferencijabilna u prijelaznoj točki, u negativnoj domeni teži $-\alpha$ pa se njezina derivacija postupno se smanjuje prema nuli. Iako se time uvodi zasićenje, ono se javlja samo za negativne vrijednosti i određeno je parametrom α , za razliku od sigmoidnih funkcija kod kojih se zasićenje pojavljuje s obje strane. Ovo ponašanje smanjuje rizik „mrtvih neurona“ i ograničava utjecaj velikih negativnih vrijednosti.

Strukturalna homogenost obitelji

Ostale funkcije unutar ove obitelji mogu se promatrati kao varijacije osnovne eksponencijalne funkcije. Primjerice, Softplus se definira kao:

$$f(x) = \ln(1 + e^x) \quad (2.10)$$

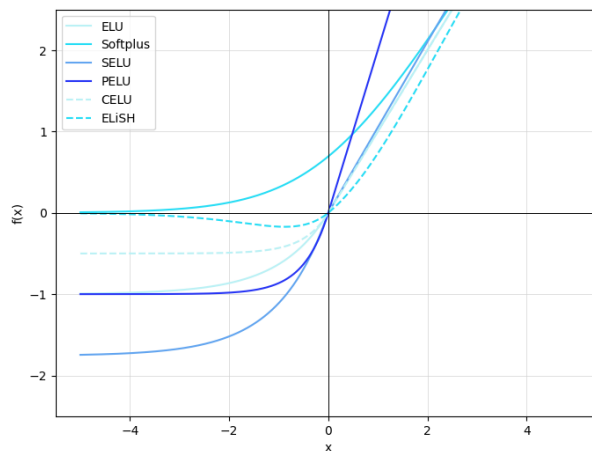
i interpretira se kao glatka aproksimacija ReLU funkcije u kojoj eksponencijalna struktura osigurava potpunu diferencijabilnost. Za razliku od ELU-a, Softplus ne uvodi negativno zasićenje već održava pozitivan izlazni raspon.

Tablica 2.3 Sažetak eksponencijalnih aktivacijskih funkcija.

Naziv	Kratice	Godina	Formula	Raspon izlaza
Softmax [64]	-	1990	$\frac{e^{z_i}}{\sum_j e^{z_j}}$	[0, 1]
Eksponencijalna linearna jedinica [65]	ELU	2016	$\begin{cases} x, & x > 0 \\ \alpha(e^x - 1), & x \leq 0 \end{cases}$	[-1, ∞)
Softplus	-	-	$\ln(1 + e^x)$	[0, ∞)
Skalirana ELU [66]	SELU	2017	$\lambda \times \begin{cases} x, & x > 0 \\ \alpha(e^x - 1), & x \leq 0 \end{cases}$	[- λ , ∞)
Parametrijska ELU [22]	PELU	2017	$\begin{cases} \frac{a}{b} \times x, & x \geq 0 \\ a \times (e^{\frac{x}{b}} - 1), & x < 0 \end{cases}$	[a, ∞)
Kontinuirano diferencijabilna ELU [67]	CELU	2017	$\begin{cases} x, & x \geq 0 \\ \alpha \times (e^{\frac{x}{\alpha}} - 1), & x < 0 \end{cases}$	[- α , ∞)
Višestruka parametrijska ELU [68]	MPELU	2018	$\begin{cases} x, & x > 0 \\ \alpha_c \times (e^{\beta_c x} - 1), & x \leq 0 \end{cases}$	[- α_c , ∞)
Eksponencijalno linearno sigmoidno sabijanje [69]	ELiSH	2018	$\begin{cases} x/(1 + e^{-x}), & x \geq 0 \\ (e^x - 1)/(1 + e^{-x}), & x < 0 \end{cases}$	[0, ∞)
Tvrđi ELiSH [69]	HardELish	2018	$\max\left(0, \min\left(1, \frac{x+1}{2}\right)\right)$	[0, ∞)
Parametrijski ispravljena eksponencijalna jedinica [70]	PREU	2019	$\begin{cases} \alpha \times x, & x > 0 \\ \alpha \times x \times e^{\beta \times x}, & x \leq 0 \end{cases}$	[-1, ∞)
Brza ELU [71]	FELU	2019	$\begin{cases} x, & x > 0 \\ \alpha \times (e^{\frac{x}{\ln(2)}} - 1), & x \leq 0 \end{cases}$	[- α , ∞)

Elastična ELU [72]	EELU	2020	$\begin{cases} k \times x, & x > 0 \\ \alpha \times (e^{\beta \times x} - 1), & x \leq 0 \end{cases}$	$[-\alpha, \infty)$
Parametrijska deformabilna ELU [73]	PDELU	2020	$\begin{cases} x, & x > 0 \\ \alpha \times ([1 + (1 - t) \times x]^{\frac{1}{1-t}} - 1), & x \leq 0 \end{cases}$	$[-1, \infty)$
Gompertzova linearna jedinica [74]	GoLU	2025	$x(e^{-e^{-x}})$	$(m, +\infty), m \approx$ ispod 0
Prošireni integral eksponencijalne linearne jedinice [75]	xIELU	2025	$\begin{cases} \alpha_p x^2 + 0.5x, & x > 0 \\ \alpha_n(e^x - 1) - \alpha_n x + 0.5x, & x \leq 0 \end{cases}$	$(m, \infty),$ $m < 0$

Unatoč razlikama, osnovna ideja je ista: funkcija je linearna za pozitivne vrijednosti, prijelaz je gladak, a izbjegava se nagli lom kakav ReLU ima. Razlike su uglavnom kvantitativne, oblik prijelaza, brzina približavanja granici i amplituda u negativnom području (Tablica 2.3, Slika 2.5).



Slika 2.5 Eksplozivne aktivacijske funkcije.

Eksplozivne aktivacijske funkcije predstavljaju kompromis između ReLU i sigmoidnih funkcija: zadržavaju optimizacijsku stabilnost ReLU-a, ali uvode glatkoću i kontrolirano zasićenje. Izbor između ELU-a, Softplusa ili srodnih funkcija, prilagodba je stupnja glatkoće i intenziteta negativne modulacije, a ne promjena tipa nelinearnosti. Softmax se često koristi u izlaznom sloju za višeklasnu klasifikaciju, gdje je cilj klasificirati ulaz u jednu od međusobno isključivih klasa, preporučuje se u kombinaciji s funkcijom gubitka križne entropije. ELU i srodne funkcije preporučuju se u dubokim skrivenim slojevima kada je stabilnost gradijenta prioritet [43].

Adaptivne aktivacijske funkcije

Nakon razvoja funkcija poput ReLU i eksponencijalnih funkcija, istraživanja su se usmjerila na aktivacijske funkcije koje se mogu prilagođavati tijekom treniranja. Za razliku od klasičnih funkcija s unaprijed zadanim oblikom, ove funkcije imaju dodatne parametre koji se uče zajedno s modelom. Samim time, ove funkcije ne uvode novi tip nelinearnosti, već samo prilagođavaju postojeće oblike, najčešće ReLU ili eksponencijalne funkcije. Time se povećava fleksibilnost modela bez promjene njegove osnovne strukture. Neki od već opisanih modela, poput PReLU i PTELU, već imaju takva svojstva.

Strukturalna homogenost obitelji

Strukturno gledano, adaptivne aktivacijske funkcije mogu se shvatiti kao proširenja postojećih funkcija, poput ReLU ili eksponencijalnih funkcija, pri čemu se njihov oblik dodatno prilagođava tijekom učenja. Osnovni način uvođenja nelinearnosti pritom ostaje isti, ne uvodi se nova vrsta transformacije, već se postojeći oblik prilagođava podacima.

Uvođenje dodatnih parametara omogućuje modelu veću fleksibilnost: različiti slojevi mogu koristiti različite oblike funkcija, a ponašanje se može bolje prilagoditi podacima pa se u nekim slučajevima ublažavaju problemi s učenjem. S druge strane, to donosi i nedostatke, poput većeg broja parametara, rizika od prenaučivosti i složenijeg treniranja. Ipak, adaptivne funkcije ne uklanjaju ograničenja osnovnih funkcija iz kojih su izvedene. Primjerice, parametrizirana ReLU i dalje ostaje dijelom linearna i ne postaje potpuno glatka funkcija pa zbog toga adaptivne funkcije ne čine jedinstvenu skupinu poput sigmoidnih ili ReLU funkcija. Njihova zajednička značajka je da se oblik uči iz podataka, dok osnovna struktura ostaje ista. Razlike su uglavnom u broju i tipu parametara (Tablica 2.4).

Tablica 2.4 Sažetak adaptivnih aktivacijskih funkcija.

Naziv	Kratica	Godina	Formula	Raspon izlaza
Po dijelovima adaptivna linearna jedinica [76]	APL	2015	$\max(0, x) + \sum_{s=1}^S a_s \times \max(0, b_s - x)$	$[0, \infty)$
Spline [77]	SAF	2016	$(\Phi s; q)$	$(-\infty, \infty)$
Bimodalna derivacija adaptivne aktivacije [78]	BDAA	2017	$\frac{1}{2} \times \left(\frac{1}{1 + e^{-x}} - \frac{1}{1 + e^{-x-a}} \right)$	$[0, 1]$
Adaptivna [79]	AAF	2018	$\sigma(w \times x) \times PRELU(x) + (1 - \sigma(w \times x)) \times PELU(x)$	$[0, 1]$
Swish [80]	-	2018	$x \times Sigmoid(\beta \times x)$	$(-\infty, \infty)$
ESwish [81]	-	2018	$\beta \times x \times Sigmoid(x)$	$(-\infty, \infty)$
Swish s T-pomakom [82]	FTS	2018	$\begin{cases} \frac{x}{1 + e^{-x}} + T, & x \geq 0 \\ T, & x < 0 \end{cases}$	$[T, \infty)$
Adaptivna aktivacija s ponderiranom Richardovom krivuljom [83]	ARiA	2018	$x \times \sigma(L)$	$(-\infty, \infty)$
Obučljiva AF [84]	TAF	2018	$\sqrt{(x - a)^2 + b^2}$	$[b, \infty)$
Samoučeća AF [85]	SLAF	2020	$\sum_{i=0}^{N-1} a_i \times x^i$	$(-\infty, \infty)$
Meksički ReLU [86]	MeLU	2020	$PRELU(x) + \sum_{j=1}^k c_j \times \max(\lambda_j - x - a , 0)$	$(-\infty, \infty)$
ErfReLU [87]	-	2023	$\begin{cases} x, & x \geq 0 \\ \alpha \operatorname{erf}(x), & x < 0 \end{cases}$	$(-\alpha, \infty)$
Reparametrizirajuća adaptivna aktivacijska funkcija [88]	RepAct	2024	$\begin{cases} x\delta_1, & x \leq 3 \\ x^2\delta_2 + x\delta_3, & 0 \leq x < 3 \\ x^2\delta_2 + x\delta_4, & -3 \leq x < 0 \\ x\delta_5, & x < -3 \end{cases}$	$(-\infty, \infty)$

Adaptivne varijante primjenjuju se u dubokim mrežama s problemom neaktivnih neurona i pri eksperimentalnoj optimizaciji performansi. Budući da izbor aktivacijske funkcije često ovisi o specifičnosti zadatka i arhitekture, naglašava se važnost empirijskog testiranja [43].

2.3 Ograničenja klasičnih modela

Iako su opisane aktivacijske funkcije omogućile uspjehe u području dubokog učenja, one su pojednostavljen prikaz stvarnih bioloških procesa. *Klasični* umjetni neuroni obrađuju ulazne podatke u jednom koraku, a izlaz ovisi samo o trenutnoj vrijednosti ulaza. U takvom pristupu vrijeme nije zasebno modelirano, a informacija se prenosi kroz razinu aktivacije neurona, čime se zanemaruje vremenska dinamika prisutna u živčanim sustavima [89], [90].

Ova ograničenja postaju vidljiva kod zadataka koji zahtijevaju energetska učinkovitost ili obradu u stvarnom vremenu. Biološki neuroni ne prenose informacije kontinuiranim vrijednostima, već kratkim električnim impulsima (engl. *spikes*) koji se javljaju u točno određenim vremenskim trenucima [89], [91]. Takav način rada omogućuje mozgu vrlo nisku potrošnju energije jer neuroni troše energiju samo kada šalju impuls, dok su ostatak vremena neaktivni [92], [93].

Prijelaz s aktivacijskih funkcija na biološki inspirirane procese zahtijeva zamjenu jednostavnih nelinearnih funkcija dinamičkim sustavima opisanim diferencijalnim jednadžbama [5]. Dok aktivacijske funkcije opisuju *što* neuron izračunava, biološki modeli opisuju *kako* se potencijal membrane mijenja kroz vrijeme pod utjecajem ionskih struja. Uvođenjem vremenske komponente, neuron prestaje biti jedinica za trenutnu obradu i postaje akumulator informacija jer integrira ulazne podražaje tijekom vremena sve dok njegov potencijal membrane ne dosegne kritični prag, nakon čega generira impuls i resetira svoje stanje.

U tom kontekstu nameće se pitanje hijerarhije biološke vjernosti. Aktivacijske funkcije optimizirane su za učinkovitu obradu velikih skupova podataka, ali uz gotovo potpuno žrtvovanje neurofiziološke preciznosti. S druge strane, biofizički vjerniji modeli nude maksimalnu biološku točnost opisujući dinamiku ionskih kanala, ali visok računalni trošak čini ga teško skalabilnim za velike mreže.

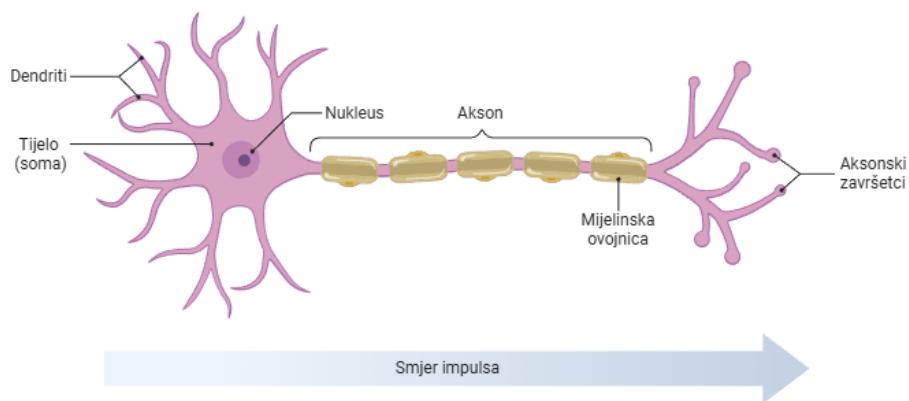
Jaz između matematičke jednostavnosti i biološke kompleksnosti definira suvremena istraživanja neuromorfog računarstva. Promatranje neurona kao dinamičkog sustava, a ne samo kao prijenosnika signala, otvara nove načine učenja, poput učenja koje ovisi o vremenskom odnosu impulsa (engl. *Spike-Timing-Dependent Plasticity*, STDP) ili Hebbovog učenja [80], [85]. Time se otvara put prema sustavima koji, poput ljudskog mozga, ne uče samo na temelju pogreške u izlazu, već na temelju precizne vremenske sinkronizacije događaja. U poglavljima koja slijede detaljnije će se analizirati kako različiti biofizički vjerniji modeli neurona nastoje balansirati između ove dvije krajnosti [95].

3 Biološki modeli neurona

Biofizički vjerniji modeli neurona predstavljaju pokušaj formalizacije neuronske obrade informacija na temelju opaženih principa živčanog sustava, uz jasno definirane kompromise između vjernosti i računalne izvedivosti. Ovakvi modeli nastoje obuhvatiti osnovne mehanizme neuronske signalizacije na način koji je bliži biološkoj stvarnosti. U skladu s time, oblikovan je niz neuronskih modela koji se razlikuju po razini apstrakcije, matematičkom opisu i računalnoj složenosti, pri čemu svaki od njih teži specifičnoj ravnoteži između biološke realističnosti i praktične primjenjivosti. U nastavku se razmatra biološki model neurona i biofizički vjerniji *umjetni* modeli neurona koji prenose osnovne principe obrade informacija u računalne sustave.

3.1 Anatomija neurona

Neuron, temeljna jedinica živčanog sustava, je specijaliziran za prijenos informacija putem elektrokemijskih signala. Neuron koji nosi informaciju naziva se presinaptički, a primatelj se naziva postsinaptički neuron. Smatra se da u ljudskom tijelu ima oko 10^{11} neurona, a jedan neuron može uspostaviti veze s više od 10^4 neurona. Ove veze mogu biti u neposrednoj blizini, ili se protezati nekoliko centimetara [89].



Slika 3.1 Struktura neurona.

Dendriti su kratki produžeci neurona koji djeluju kao ulazni kanali za signale koji dolaze u obliku neurotransmitera (kemijski signali) koje oslobađaju drugi neuroni i pretvaraju se u električne signale. Soma, ili tijelo stanice, integrira te signale kako bi odredila hoće li neuron pokrenuti akcijski potencijal, koji se zatim prenosi na akson. Ovaj proces je poznat kao neuronska integracija. Akson prenosi akcijski potencijal drugim neuronima. Kako bi se akcijski potencijal brzo prenio na velike udaljenosti bez slabljenja, neki aksoni su obavijeni mijelinskom ovojnicom [96]. Aksonski završeci su krajnji dijelovi aksona koji služe za prijenos signala prema drugoj stanici. Prostor gdje se informacije prenose a neuroni „spajaju“ zove se sinapsa, a one mogu biti kemijske ili električne [97]. U kemijskim sinapsama nema izravnog kontakta između neurona. Signal iz neurona prenosi se preko neurotransmitera sadržanih u sinaptičkim granulama otpuštenim u sinaptičku pukotinu čime se mijenja potencijal membrane. Ova vrsta prijenosa je sporija, ali može pojačati signal i produžiti njegovo trajanje. Kemijske sinapse mogu biti ekscitatorne, koje depolariziraju postsinaptičku stanicu (imaju tendenciju da generiraju električni impuls) i potiču akcijske potencijale, ili inhibitorne, koje hiperpolariziraju stanicu (imaju tendenciju da spriječe neuron od generiranja električnog impulsa) i sprječavaju akcijske potencijale. S druge strane, električne sinapse su strukture koje prenose naboje membrane do sljedećeg neurona preko praznih spojeva. Ova vrsta komunikacije je vrlo brza,

međutim, nema povećanja amplitude signala kao u kemijskim sinapsama. Električne sinapse brojčano nadmašuju kemijske sinapse u živčanom sustavu u razvoju, a kemijske sinapse brojčano nadmašuju električne sinapse u potpuno razvijenom živčanom sustavu.

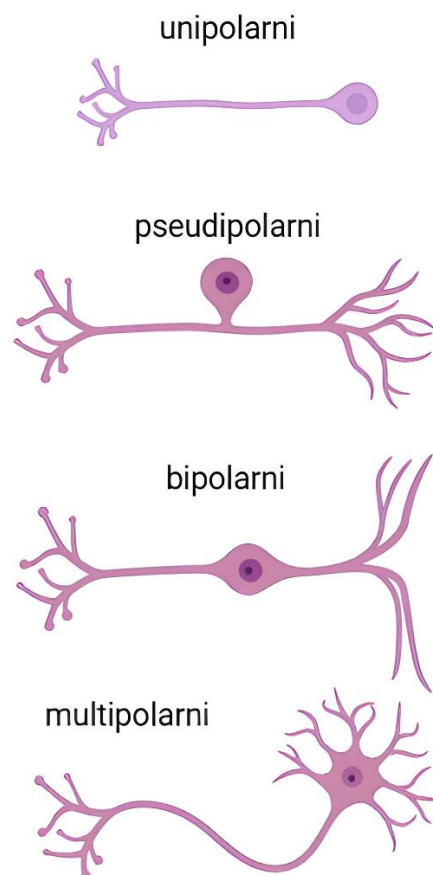
3.2 Klasifikacija neurona

Iako se sve živčane stanice kolektivno nazivaju neuronima, njihova morfologija, veličina i funkcionalna uloga značajno variraju. U neuroznanosti, klasifikacija neurona predstavlja dugotrajan izazov, prvenstveno zbog poteškoća u definiranju taksonomskih granica. S jedne strane, nepobitno je da je svaki neuron jedinstven u svom mikrookruženju, no preveliko uvažavanje individualnih razlika učinilo bi koncept „tipa neurona“ neupotrebljivim. S druge strane, široke kategorizacije su često nedovoljno precizne za specifične eksperimentalne svrhe [98].

U kontekstu dinamike prijenosa signala, primarna podjela temelji se na relativnom položaju neurona u odnosu na sinapsu. Razlikujemo presinaptički neuron (odašiljač) i postsinaptički neuron (prijemnik). Presinaptički neuron je smješten prije sinapse i on pretvara električne signale (akcijske potencijale) u kemijske glasnike (neurotransmitere) koje otpušta u sinaptičku pukotinu. Postsinaptički neuron, smješten nakon sinapse, prima neurotransmitere i inicira novi signal. Ove uloge su relativne u odnosu na sinapsu, što znači da jedan neuron može biti i pre- i postsinaptički, ovisno o smjeru signalizacije.

Strukturna raznolikost neurona odražava njihovu specijalizaciju u prijenosu i obradi informacija. Prema broju i rasporedu izdanaka koji polaze iz staničnog tijela (some), neurone klasificiramo u četiri osnovna morfološka tipa (Slika 3.2). Unipolarne neurone karakterizira samo jedan izdanak koji se pruža od some. Ovi su neuroni tipični za beskralježnjake, dok se u kralježnjaka u ovom obliku gotovo ne pojavljuju. Bipolarni neuroni posjeduju jedan akson i jedan dendrit koji polaze s suprotnih polova some. Često su specijalizirani za prijenos osjetilnih informacija, poput onih u mrežnici oka. Multipolarni neuroni su najzastupljeniji tip neurona u središnjem živčanom sustavu. Sadrže jedan akson i veći broj razgranatih dendrita koji omogućuju integraciju informacija iz brojnih izvora. Primjer multipolarnog neurona je Purkinjeova stanica u malom mozgu sa složenim i gustim stablom dendrita povezanih sa samo jednim aksonom. Pseudounipolarni neuroni su morfološki hibridi; iz some polazi jedan zajednički produžetak koji se ubrzo grana u dvije zasebne strukture. Na taj način jedna

grana djeluje kao dendrit (prijemnik), dok druga preuzima ulogu aksona [99], [100].



Slika 3.2 Klasifikacija neurona po strukturi.

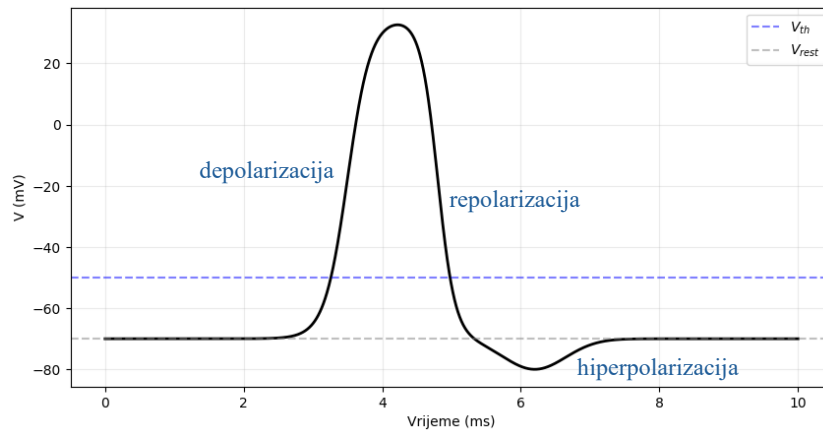
Zatim, neurone klasificiramo prema utjecaju na druge stanice u dvije primarne kategorije: ekscitatorne i inhibitorne. Ekscitatorni neuroni povećavaju vjerojatnost izbijanja akcijskog potencijala kod postsinaptičkog neurona uzrokujući depolarizaciju membrane. Suprotno, inhibitorni neuroni smanjuju tu vjerojatnost kroz proces hiperpolarizacije. Potrebno je razlikovati strukturne uloge (pre / postsinaptički) od funkcionalnog karaktera (ekscitatorni / inhibitorni) neurona. Dok prvi opisuju isključivo smjer toka informacije kroz sinapsu, drugi definiraju biološki učinak na ciljni neuron kroz modulaciju potencijala membrane [101]. Presinaptički neuron može biti ekscitatoran ili inhibitoran, no njegov konačni učinak ne ovisi isključivo o otpuštenom neurotransmiteru, već i o specifičnim receptorima na postsinaptičkoj membrani. U biološkim sustavima, jedan postsinaptički neuron obično prima i integrira više ekscitatornih i inhibitornih ulaza istovremeno čime se oblikuje njegova aktivnost.

Alternativna podjela temelji se na fiziološkoj ulozi koju neuroni obavljaju unutar organizma, pri čemu razlikujemo tri osnovne klase. Senzorni (osjetilni) neuroni specijalizirani su za prikupljanje informacija iz unutarnjeg i vanjskog okruženja te njihov prijenos prema središnjem živčanom sustavu. Zanimljivo je da je većina senzornih neurona pseudounipolarnog tipa, što im omogućuje učinkovit prijenos vanjskih signala izravno u leđnu moždinu, bez prethodne obrade u somi. Motorički neuroni primaju signale od drugih neurona te prenose informacije mišićima, organima i žlijezdama. Morfološki su to primarno multipolarni neuroni, što im omogućuje integraciju velikog broja signala prije same aktivacije mišića. Interneuroni se nalaze samo u središnjem živčanom sustavu i služe kao posrednici koji povezuju senzorne, motoričke i druge interneurone. Oni čine najbrojniju klasu neurona i ključni su za procesiranje informacija i više kognitivne funkcije [99], [102]. Kao i motorički, interneuroni su većinom multipolarni.

3.3 Akcijski potencijal i refraktornost

Temeljna jedinica komunikacije u živčanom sustavu je akcijski potencijal, u literaturi često nazivan impuls ili šiljak. On predstavlja brzu, kratkotrajnu i samopropagirajuću promjenu električnog potencijala na membrani neurona koja se širi duž aksona prema sinaptičkim terminalima. U stanju mirovanja, neuron održava potencijal mirovanja koji iznosi približno -70 mV . Ovaj potencijal rezultat je razlike u koncentraciji iona, izvan stanice koncentracija natrijevih iona (Na^+) je viša, a unutra stanice viša je koncentracija kalijevih iona (K^+). Potencijal membrane potencijal nije statičan već kontinuirano oscilira ovisno o podražajima s aksona susjednih neurona. Ekscitatorni ulazi povećavaju potencijal čime potiču generiranje impulsa, dok inhibitorni ulazi potencijal čine negativnijim i sprječavaju aktivaciju neurona [95]. Proces nastanka akcijskog potencijala odvija se kroz nekoliko faza [89], prikazano na Slika 3.3 Slika 3.4. Proces započinje ulaskom pozitivno nabijenih iona u tijelo stanice putem kemijski upravljanih kanala. Aktivacija navedenih kanala rezultat je vezivanja specifičnih neurotransmitera za postsinaptičke receptore lokalizirane na dendritima, što dovodi do lokalne promjene potencijala. Ukoliko akumulirani naboj dosegne kritični prag podražaja, na bazi aksona dolazi do otvaranja velikog broja naponski upravljanih natrijevih kanala, čime neuron prelazi u fazu nagle depolarizacije. Tijekom depolarizacije, ulazak iona Na^+ u unutrašnjost aksona uzrokuje promjenu polariteta membrane, pri čemu unutarstanični prostor poprima pozitivan električni naboj u odnosu na izvanstanični medij. Propagacija akcijskog potencijala odvija se kontinuirano i sekvencijalno duž membrane aksona, pri čemu depolarizacija jednog segmenta membrane inducira aktivaciju susjednih, naponom ovisnih kanala, osiguravajući nesmanjeni prijenos signala. Nakon dosezanja maksimalne vrijednosti potencijala, slijedi faza repolarizacije u kojoj se natrijevi kanali zatvaraju i prekida se daljnji ulazak pozitivnog naboja. Istovremeno, otvaranje kalijevih kanala omogućuje izlazak K^+ iona, što rezultira vraćanjem vrijednosti potencijala membrane u mirovanja. Budući da kalijevi kanali ostaju otvoreni nešto duže nego što je potrebno za uspostavu osnovne ravnoteže, sustav ulazi u fazu hiperpolarizacije

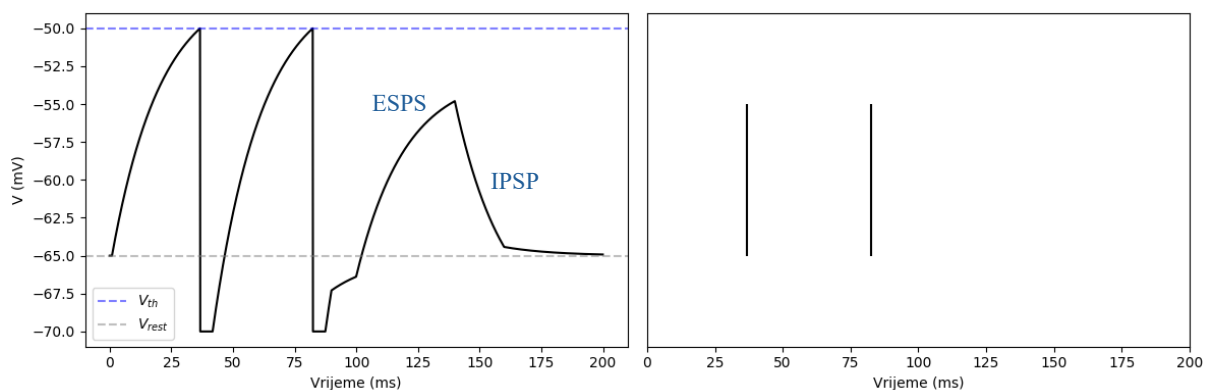
u kojoj stanica privremeno postaje električki negativnija u odnosu na svoje uobičajeno stanje mirovanja. U završnoj fazi, natrij-kalij pumpa ponovno uspostavlja početne koncentracije pa se vraća u stanje pripravnosti za obradu sljedećeg impulsa.



Slika 3.3 Shematski prikaz potencijala membrane tijekom akcijskog potencijala. Krivulja predstavlja idealizirani prikaz promjene potencijala membrane kroz faze depolarizacije, dosezanja maksimalne vrijednosti, repolarizacije i naknadne hiperpolarizacije.

Komunikacija između neurona temelji se na diskretnim događajima; akcijski potencijal se generira tek kada suma svih ekscitatornih i inhibitornih ulaza dovede potencijal membrane do praga okidanja (obično oko -50 mV). Proces kojim neuron integrira i obrađuje signale naziva se sinaptička sumacija (Slika 3.4).

Pojedinačni ekscitatorni postsinaptički potencijal (engl. *Excitatory postsynaptic potential*, EPSP) u pravilu nije dovoljan za okidanje neurona. Međutim, vremenska i prostorna akumulacija više EPSP-ova, nastalih u kratkom vremenskom razdoblju ili na različitim granama dendrita, može rezultirati depolarizacijom dovoljnom za dosezanje praga okidanja. Suprotno, inhibitorni postsinaptički potencijali (engl. *Inhibitory postsynaptic potential*, IPSP) smanjuju vjerojatnost nastanka akcijskog potencijala hiperpolarizacijom membrane tako da se potencijal membrane udaljava od praga okidanja. Ovakav mehanizam djeluje kao biološki filter koji osigurava da se šum u ulaznim signalima ne prenose dalje kao važna informacija.



Slika 3.4 Simulacija potencijala membrane neurona pomoću Leaky integrate-and-fire modela. Lijevo, pobudna struja djeluje u intervalima 1-90 ms (18 nA) i 100-140 ms (12 nA) sa refraktornim razdobljem (5 ms), dok inhibicijska struja djeluje u intervalu 140-160 ms (-5 nA). Desno su prikazani impulsi generirani nakon što potencijal membrane dosegne prag okidanja.

Akcijski potencijali slijede načelo „sve ili ništa”. Neuron nakon dosezanja praga okidanja generira impuls približno jednakog oblika i trajanja zato se informacija ne prenosi jačinom impulsa, već njihovom učestalošću (engl. *firing frequency*) gdje jači podražaji stvaraju više impulsa.

Maksimalna učestalost generiranja akcijskih potencijala ograničena je refraktornim razdobljem koje osigurava diskretnu prirodu neuronske signalizacije i definira vremenska ograničenja. Refraktorno razdoblje sastoji se od dvije funkcionalno različite faze [89]. Apsolutno refraktorno razdoblje obuhvaća kratki interval, u trajanju od približno 1-2 ms, neposredno nakon nastanka akcijskog potencijala, tijekom kojeg je fizički nemoguće generirati novi akcijski potencijal, neovisno o jačini podražaja. Relativno refraktorno razdoblje nastupa nakon apsolutnog i podudara se s fazom hiperpolarizacije membrane. Iako su ionski kanali ponovno funkcionalni, za doseganje praga depolarizacije potreban je snažniji ekscitatorni podražaj u odnosu na fiziološke uvjete mirovanja. Refraktorno razdoblje ne ograničava samo brzinu okidanja neurona, već daje vrijeme za oporavak.

3.4 Vrijeme kao nositelj informacije i strategije kodiranja

Mozak sisavaca sadrži preko 10^{10} gusto raspoređenih neurona, povezanih u složenu strukturu koji emitiraju tisuće akcijskih potencijala u svakoj milisekundi. Ova činjenica prirodno otvara jedno od temeljnih pitanja suvremene neuroznanosti: na koji način mozak kodira i prenosi informaciju putem diskretnih električnih impulsa [103]? Drugim riječima, kako se kontinuirani vanjski podražaji prevode u nizove akcijskih potencijala i kako se iz tih nizova ponovno rekonstruira značenje?

Za razliku od klasičnih računalnih sustava u kojima se informacija obrađuje kroz niz brojevnih vrijednosti, u živčanom sustavu informacija je neraskidivo vezana s vremenom. Značenje signala ne leži u amplitudi pojedinog akcijskog potencijala, već u vremenskom obrascu njihova pojavljivanja. Unatoč desetljećima istraživanja, ne postoji jedinstvena teorija neuronskog kodiranja³ koja bi u potpunosti objasnila način na koji neuroni reprezentiraju i prenose informaciju.

Pristupi neuronskom kodiranju mogu se grubo podijeliti u dvije osnovne kategorije: kodiranje učestalošću (engl. *rate coding*) i vremensko kodiranje (engl. *temporal coding*) [104]. Razlika među njima proizlazi iz pitanja je li za prijenos informacije presudan točan trenutak pojave impulsa ili je dovoljna njihova prosječna frekvencija.

Povijesno najznačajniji i najkorišteniji pristup je kodiranje učestalošću prema kojem se informacija prenosi prosječnom frekvencijom izbijanja impulsa unutar određenog vremenskog intervala. Uz njega postoje i drugi načini kodiranja, koji se koriste ovisno o problemu i načinu promatranja sustava. Iako je kodiranje učestalošću uspješno u mnogim slučajevima, može biti nedostatan za objašnjenje vrlo brzine ljudske percepcije. Zbog toga se uvodi vremensko kodiranje u kojem je važan točan trenutak pojavljivanja impulsa, a ne samo njihov broj. Ovi pristupi se mogu promatrati kao različiti načini na koje se informacija predstavljena i interpretirana kroz neuronsku aktivnost [89].

Struktura pristupa neuronskom kodiranju koji se obrađuju u ovom poglavlju može se sažeto prikazati na sljedeći način:

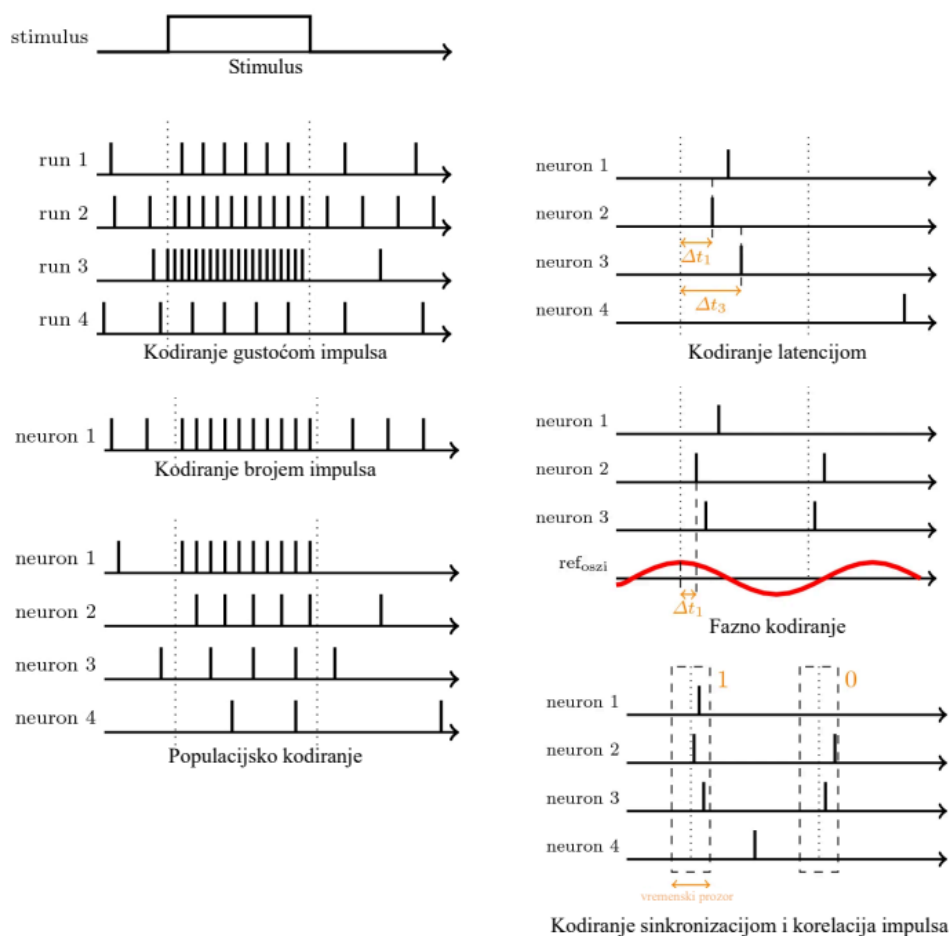
Kodiranje učestalošću:

- prosjek kroz vrijeme
- prosjek kroz ponavljanja
- prosjek kroz populaciju

Vremensko kodiranje:

- kodiranje latencijom
- fazno kodiranje
- sinkronizacija i korelacija impulsa

³ Neuronsko kodiranje (ili neuronska reprezentacija) opisuje odnos između podražaja i odgovora neurona te prijenos informacija unutar neuronskih mreža.



Slika 3.5 Vizualizacija tehnika kodiranja. Slika izmijenjena prema [104].

3.4.1 Kodiranje učestalošću

Kodiranje učestalošću (engl. *rate coding*) temelji se na pretpostavci da je informacija o podražaju sadržana u prosječnoj frekvenciji okidanja neurona. Frekvencija okidanja može se procijeniti na različite načine, ovisno o kontekstu promatranja i dostupnim podacima. Najčešće se razlikuju prosjek kroz vrijeme, kroz ponavljanja istog podražaja te kroz aktivnost populacije neurona.

Prosijek kroz vrijeme

Najjednostavniji i najčešće korišten način procjene frekvencije okidanja temelji se na promatranju aktivnosti jednog neurona unutar konačnog vremenskog intervala. U tom slučaju frekvencija se definira kao broj akcijskih potencijala $n_{sp}(T)$ koji se javljaju unutar vremenskog prozora T , podijeljen s trajanjem tog intervala:

$$v = \frac{n_{sp}(T)}{T} \quad (3.1)$$

pri čemu se rezultat obično izražava u jedinicama s^{-1} ili Hz [89], [105].

Ovakav pristup predstavlja osnovu kodiranja učestalošću i koristi se više od sto godina u istraživanjima senzorskih i motoričkih sustava, gdje se broj impulsa povećava s jačinom podražaja. Pretpostavlja se da se relevantna informacija može dobiti integracijom većeg broja impulsa, pri čemu pojedinačni trenuci njihovog pojavljivanja nisu presudni. Međutim, takav

pristup zanemaruje informaciju koja može biti sadržana u točnom vremenu pojavljivanja impulsa. Osim toga, brze reakcije organizma, koje traju svega nekoliko stotina milisekundi, otežavaju pouzdanu procjenu frekvencije temeljene na prosjeku većeg broja impulsa. U stvarnim uvjetima, ulazni signali rijetko su stabilni; čak i pri promatranju statične slike male promjene pogleda uzrokuju stalne promjene signala na mrežnici.

Unatoč tim ograničenjima, kodiranje učestalošću i dalje se široko koristi. U eksperimentima se aktivnost neurona često ispituje primjenom konstantne struje. Odnos između struje i učestalosti impulsa opisuje se tzv. $f-I$ krivuljom. U modelima se taj odnos zapisuje kao $v = g(I_0)$, gdje funkcija g opisuje kako neuron reagira na ulaznu struju. U ovom pristupu, impulsi se promatraju kao način prijenosa informacije, dok se varijacije u njihovom točnom vremenu pojavljivanja često modeliraju kao šum. Kako bi se dobila pouzdana procjena učestalosti, potrebno je promatrati veći broj impulsa i izračunati prosjek.

Prosjek kroz ponavljanja

Jedan od načina procjene frekvencije okidanja temelji se na prosjeku kroz višestruka ponavljanja istog podražaja. Takav pristup često se koristi u eksperimentalnim uvjetima kada se promatra odgovor neurona na vremenski promjenjive ili stabilne podražaje [89], [105]. Promatra se broj impulsa unutar određenog vremenskog intervala, koji se zatim uprosječuje kroz više ponavljanja, čime se dobiva pouzdanija procjena aktivnosti neurona.

Ipak, ovakav pristup nije uvijek biološki realističan, jer se mnoge odluke donose na temelju jednog podražaja. S druge strane, uprosječivanje ima smisla ako velik broj neurona istovremeno prima isti ulaz. U tom slučaju, prosjek kroz populaciju neurona može zamijeniti ponavljanje istog eksperimenta više puta. Zbog toga se u modeliranju često koristi prosjek preko populacije neurona kao alternativni način procjene frekvencije [89].

Prosjek kroz populaciju

Populacijsko kodiranje može se promatrati kao proširenje frekvencijskog kodiranja, u kojem se informacija ne izvlači iz aktivnosti jednog neurona, već iz aktivnosti skupine neurona. Biološki živčani sustavi u velikoj se mjeri oslanjaju na takav pristup, gdje se informacija interpretira kroz kolektivnu aktivnost populacije [89], [105]. Istraživanja pokazuju da velik broj neurona u mozgu ima slična receptivna svojstva i reagira na iste ili vrlo srodne podražaje. U idealiziranom modelu, razmatra se populacija neurona s istim funkcionalnim svojstvima i identičnim obrascima ulaznih i izlaznih veza. Akcijski potencijali neurona iz presinaptičke populacije šalju se prema postsinaptičkoj populaciji, pri čemu svaki neuron u postsinaptičkoj populaciji prima ulaz od svih neurona iz presinaptičke skupine. Iz perspektive postsinaptičkih neurona nije bitno „tko“ je izvor, već „koliko“ je aktivnih neurona unutar presinaptičke populacije. Formalno, aktivnost populacije $A(t)$ može se definirati kao:

$$A(t) = \frac{1}{\Delta t} \frac{n_{act}(t; t + \Delta t)}{N} \quad (3.2)$$

gdje n_{act} predstavlja broj zabilježenih impulsa u vremenskom intervalu Δt , a N broj neurona u populaciji.

Ova veličina predstavlja frekvenciju okidanja populacije i može se mijenjati gotovo trenutačno i odraziti promjene u podražaju čime se izbjegava ograničenje frekvencije definirane dugotrajnim vremenskim usrednjavanjem aktivnosti pojedinačnih neurona. Važno je naglasiti da je pretpostavka homogene populacije pojednostavljena jer biološke neuronske populacije uvijek pokazuju heterogenost u parametrima i povezivosti, kako u unutarnjim dinamičkim parametrima, tako i u strukturi sinaptičkih veza. Unatoč tome, aktivnost skupine neurona

pokazala se korisnim načelom kodiranja u mnogim dijelovima mozga, osobito kada se promatra na odgovarajućoj vremenskoj i prostornoj skali.

3.4.2 Vremensko kodiranje

Za razliku od kodiranja učestalošću, vremensko kodiranje temelji se na pretpostavci da informacija nije sadržana samo u broju impulsa, već i u njihovom točnom vremenskom rasporedu. Ovakav pristup omogućuje bržu obradu informacija i može objasniti situacije u kojima organizam reagira unutar vrlo kratkog vremenskog intervala.

Kodiranje latencijom

Kodiranje latencijom (engl. *time-to-first-spike*) temelji se na pretpostavci da je informacija sadržana u vremenu koje protekne od pojave podražaja do emitiranja prvog akcijskog potencijala neurona [106]. Slabiji podražaji uzrokuju kasnije aktiviranje neurona a jači podražaji rezultiraju kraćom latencijom prvog impulsa što omogućuje brz prijenos informacija. U idealiziranom modelu pretpostavlja se da svaki neuron prenosi informaciju prvenstveno putem prvog impulsa nakon pojave podražaja, dok se kasniji impulsi zanemaruju. U takvim pojednostavljenim pristupima često se pretpostavlja da neuron nakon izbijanja privremeno ne reagira na isti podražaj. U stvarnim biološkim sustavima neuroni mogu emitirati više impulsa, no prvi impuls često nosi značajan dio informacije.

Iako je ovaj pristup pojednostavljen, njegova biološka važnost je potvrđena. U mnogim perceptivnim zadacima većina informacije o novom podražaju prenosi se unutar prvih 20 do 50 milisekundi nakon početka neuronskog odgovora, što podupire važnost brzog vremenskog kodiranja.

Fazno kodiranje

Fazno kodiranje je oblik vremenskog kodiranja u kojem informacija ne ovisi samo o tome *kada* se impuls pojavi, već o njegovom *položaju* unutar ponavljajućeg ciklusa neuronskih oscilacija. Oscilacije služe kao referenca jer isti impuls može imati različito značenje ovisno o tome u kojem se dijelu ciklusa pojavljuje. Za isti podražaj, vremenski raspored impulsa u odnosu na fazu oscilacije može pokazivati određeni stupanj konzistentnosti kroz više ciklusa, što omogućuje stabilno i precizno predstavljanje informacije. Na taj način, aktivnost pojedinačnih neurona povezuje se s globalnim ritmovima aktivnosti mozga, što doprinosi koordinaciji obrade informacija između različitih regija [107], [108].

Sinkronizacija i korelacija impulsa

Osim odnosa prema vanjskim ili unutarnjim referentnim signalima, neuronski impulsi mogu nositi informaciju i kroz međusobne vremenske odnose između različitih neurona [106]. U tom slučaju, impulsi drugih neurona služe kao referenca, pa istovremeno odnosno usklađeno okidanje više neurona može označavati određene događaje ili značenja koja nisu vidljiva iz frekvencije okidanja. Jedna od poznatih hipoteza temeljenih na ovom principu jest da sinkronizacija neurona može označavati pripadnost istom objektu. Neuroni koji kodiraju različite značajke istog objekta mogu biti „označeni” sinkronim okidanjem, čime se sugerira njihova funkcionalna povezanost. Osim potpune sinkronizacije, i precizni vremenski obrasci impulsa mogu nositi značenje. Primjerice, određeni podražaj može biti predstavljen specifičnim slijedom impulsa više neurona s određenim vremenskim razmacima pa promjena tih odnosa unutar istog skupa neurona može odgovarati drugačijem sustavu stanju. Ovakav pristup dodatno naglašava važnost vremena u neuronskoj aktivnosti kao nositelja informacije.

Strategije neuronskog kodiranja mogu se promatrati kao različiti načini predstavljanja i interpretacije informacije u neuronskim sustavima. U stvarnim biološkim sustavima ovi pristupi nisu međusobno isključivi, već se često nadopunjuju. Kombinacija frekvencijskih i vremenskih mehanizama omogućuje fleksibilno, robusno i učinkovito kodiranje informacija, prilagođeno zahtjevima različitih kognitivnih i senzorskih zadataka.

3.5 Biofizički modeli neurona

Impulsni modeli neurona (engl. *Spiking neuron model*) predstavljaju matematičke opise ponašanja bioloških neurona koji su sposobni generirati diskretne električne impulse, odnosno akcijske potencijale. Impulsi se generiraju kada potencijal membrane V prijeđe definirani prag θ . Vrijeme $t^{(f)}$ označava trenutak generiranja akcijskog potencijala, odnosno vrijeme u kojem potencijal membrane doseže prag [89]:

$$t^{(f)} : V(t^{(f)}) = \theta, \quad \left. \frac{dV(t)}{dt} \right|_{t=t^{(f)}} > 0 \quad (3.3)$$

Njihova uloga u računalnoj neuroznanosti je razumijevanje načina na koji neuroni obrađuju i prenose informacije putem vremenski strukturiranih signala. Za razliku od klasičnih umjetnih neurona kod kojih je izlaz kontinuirana vrijednost, u impulsnim modelima informacija je sadržana u vremenu pojavljivanja impulsa. Impulsni modeli opisuju odnos između ulazne struje i izlazne aktivnosti neurona; ako neuron prima konstantnu struju, on generira iste akcijske potencijale, a vremenski promjenjivi ulazi rezultiraju nepravilnim obrascima okidanja. Dinamika impulsa tako odražava promjene u ulaznom signalu, čime se omogućuje vremensko kodiranje informacija. Impulsni modeli neurona mogu se podijeliti u dvije kategorije: biofizički modeli temeljeni na ionskoj vodljivosti i fenomenološki modeli temeljeni na integraciji ulaza i pragu okidanja [109], ali glavna razlika među modelima neurona proizlazi iz odnosa između razine biološke detaljnosti i računalne učinkovitosti.

Biofizički modeli temelje se na parametrima koji imaju izravnu fiziološku interpretaciju i eksperimentalnu mjerljivost. Hodgkin-Huxley [110] model koristi sustav nelinearnih diferencijalnih jednadžbi za detaljan opis ionskih struja kroz membranu i nudi najvišu biološku vjernost, ali njegova složenost rezultira značajnim računalnim zahtjevima, što ga ograničava na analizu pojedinačnih neurona ili manjih populacija. Kao kompromisno rješenje razvijeni su FitzHugh-Nagumo [111], Hindmarsh-Rose [112], Morris-Lecar [113] i Wilson [114], [115] modeli, koji zadržavaju ključne mehanizme pobude uz smanjenu matematičku kompleksnost.

Nasuprot tome, fenomenološki modeli (često nazivani „integriraj i aktiviraj“ modeli (engl. *Integrate-and-Fire*)) pojednostavljaju biološke tako da promatraju akcijski potencijal kao događaj u vremenu. Njegova promjena kroz vrijeme ovisi o ulaznim impulsima koji dolaze preko sinapsi, kao i o svojstvima membrane zbog kojih se potencijal postupno smanjuje kada nema pobude. Zbog toga neuroni mogu prikupljati informacije kroz vrijeme, jer učinak pojedinog impulsa ne nestaje odmah, već postupno slabi. Umjesto detaljnog modeliranja oblika impulsa, ovi modeli fokusiraju se na trenutak kada impuls nastaje, koristeći jednostavne jednadžbe za opis promjene potencijala membrane do praga okidanja. Zbog toga su potrebne barem dvije jednadžbe, jedna koja opisuje promjenu potencijala, a druga određuje kada dolazi do impulsa. Zbog jednostavnosti i malog broja parametara, fenomenološki modeli se češće koriste u računalnoj neuroznanosti.

3.5.1 Leaky Integrate-and-Fire model neurona

Obitelj „integriraj i aktiviraj“ modelira neuron kao sustav koji zbraja ulazne impulse. Kada potencijal membrane $V(t)$ prijeđe prag θ , neuron generira impuls i resetira se početnu

vrijednost $V \leftarrow V_{rest}$ (nakon čega opcionalno slijedi refraktorni period). U najjednostavnijem slučaju, neuron akumulira ulazne signale bez pasivnog gubitka naboja [116]. Promjena potencijala membrane opisana je jednačžbom:

$$C \frac{dV(t)}{dt} = I(t) \quad (3.4)$$

gdje C označava kapacitet membrane, a $I(t)$ ulaznu struju. Ovaj model je računalno jednostavan ali u biološkim sustavima potencijal membrane ne akumulira ulazne struje beskonačno dugo. Zbog pasivnih ionskih struja i svojstava stanične membrane, nakupljeni naboj s vremenom se spontano troši. Ovaj efekt, poznat kao curenje membrane (engl. *leak*), bitan je za stabilnost neuronske dinamike i vremenski ograničenu integraciju signala. Da bi se uveo ovaj mehanizam, idealni integrator proširuje se dodavanjem člana koji opisuje pasivno curenje membrane [117], [118] i dobiva se Leaky Integrate-and-Fire (LIF) model.

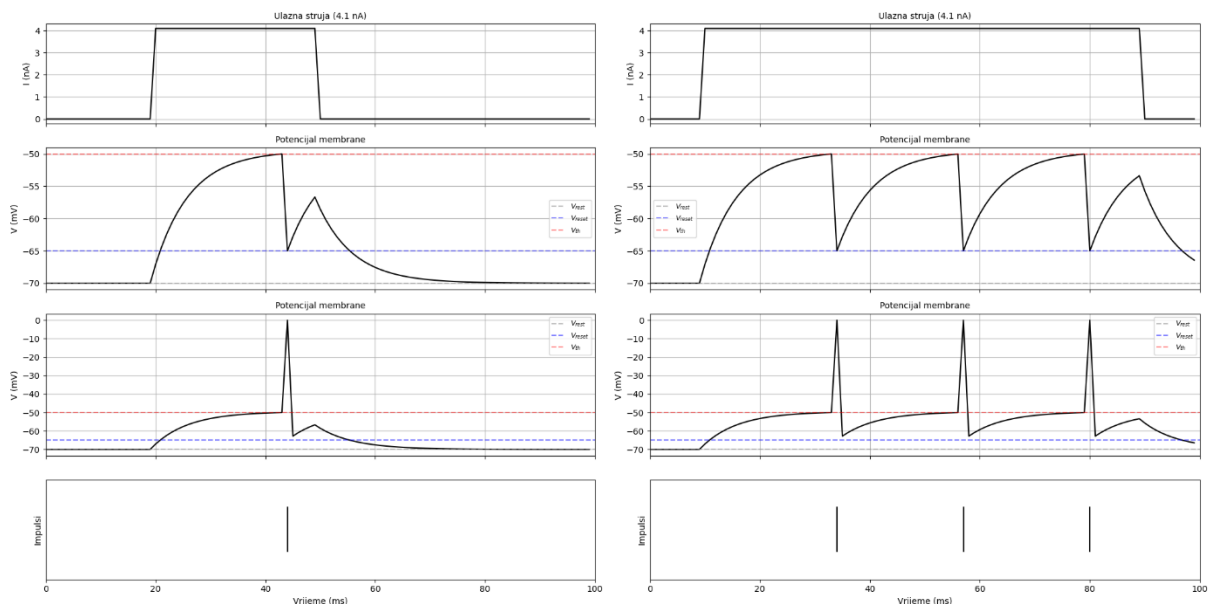
$$C \frac{dV(t)}{dt} = -g_L(V(t) - V_{rest}) + I(t) \quad (3.5)$$

U gornjoj jednačžbi g_L predstavlja provodnost curenja. Jednačžba se često zapisuje u alternativnom obliku korištenjem vremenske konstante membrane $\tau = \frac{C}{g_L}$, koja kontrolira brzinu curenja, akumulaciju struje i otpora membrane $R = \frac{1}{g_L}$, čime se dobiva zapis:

$$\tau \frac{dV(t)}{dt} = -(V(t) - V_{rest}) + RI(t) \quad (3.6)$$

Za konstantan ulaz, minimalna struja potrebna da neuron dostigne prag može se izraziti kao $I_{th} = \frac{V_{th}}{R}$. Ako pretpostavimo da se potencijal membrane resetira na 0, frekvencija okidanja koja se za velike vrijednosti ulaznog strujnog signala približava prethodnom modelu bez curenja, uz uključeno refraktorno razdoblje [119], može se izraziti kao:

$$f(I) = \begin{cases} 0, & I \leq I_{th} \\ \left[t_{ref} - RC \log \left(1 - \frac{V_{th}}{IR} \right) \right]^{-1}, & I > I_{th} \end{cases} \quad (3.7)$$



Slika 3.6 Potencijal membrane LIF neurona modeliran prema jednadžbi. Vremenska konstanta membrane postavljena je na $\tau=7$ ms, električni otpor membrane $R=5$, a je potencijal membrane u mirovanju $V_{rest}=-70$ mV. Prag okidanja neurona je $V_{th}=-50$ mV, zatim se resetira na $V_{rest}=-65$ mV. Simulacija je provedena tijekom 100 ms s vremenskim korakom $dt=1$ ms. Ulazna struja amplitude 4.1 nA, primijenjena u vremenskom intervalu od 20-50 ms i 10-90 ms. Drugi neuron ima parametar „naglasiti“ $=-0.001$, koji ne mijenja trenutak ni stopu okidanja, već služi isključivo grafičkoj reprezentaciji akcijskog potencijala.

U LIF modelu, impuls se generira onog trenutka kada potencijal membrane dosegne prag (V_{th}) (Slika 3.6). Kada se prag dosegne, napon se ručno resetira pa trenutak pojave impulsa nije posljedica jednadžbe već vanjskog pravila. Za realističniji opis ponašanja membrane, „integriraj i aktiviraj“ modeli mogu se opisati i nelinearnim jednadžbama:

$$\tau \frac{dV(t)}{dt} = F(V) + G(V)I \quad (3.8)$$

Funkcija $G(V)$ se može interpretirati kao ulazni otpor ovisan o potencijalu membrane, dok izraz $-\frac{F(V)}{V-V_{rest}}$ odgovara konstanti opadanja ovisnoj o naponu. Na toj se osnovi razvija Eksponecijalan integriraj i aktiviraj neuron (EIF), koji bolje opisuje nastanak impulsa uvođenjem eksponencijalnog člana u dinamiku membrane čime se dobiva „glatki“, biofizički utemeljen prag okidanja [120]:

$$\tau \frac{dV(t)}{dt} = -(V(t) - V_{rest}) + \Delta_T \exp\left(\frac{V(t) - V_T}{\Delta_T}\right) + RI(t) \quad (3.9)$$

gdje je V_T prag okidanja, Δ_T određuje koliko naglo započinje impuls. Impuls se generira kada $V(t)$ prijeđe unaprijed zadanu maksimalnu vrijednost (npr. V_{spike}), nakon čega se potencijal membrane resetira na V_{reset} , uz mogućnost uvođenja refraktornog razdoblja τ_{ref} . U usporedbi s LIF modelom, EIF vjernije reproducira brzi porast akcijskog potencijala te preciznije odgovore na brze, promjenjive ulaze, uz znatno manju računalnu složenost u odnosu na modele temeljene na ionskim vodljivostima [120]. Iako nelinearnost i dodatni parametri povećavaju točnost modela, zahtijevaju i podešavanje Δ_T i V_T što može otežati numeričko računanje u blizini praga.

Adaptivno-eksponencijalni integriraj i aktiviraj model (AdEx) predstavlja proširenje LIF neurona koje kombinira eksponencijalni opis početka impulsa s varijablom adaptacije ovisne o okidanju koja omogućuje različite obrasce neuronske aktivnosti uz pravilno i brzo izbijanje [121].

$$C_m \frac{dV_m}{dt} = -G_L(V_m - E_L) + G_L \Delta_T \exp\left(\frac{V_m - V_T}{\Delta_T}\right) - w + I_{syn} \quad (3.10)$$

$$\tau_w \frac{dw}{dt} = a(V_m - E_L) - w \quad (3.11)$$

Ovdje je V_m potencijal membrane, C_m kapacitet membrane, G_L vodljivost curenja, E_L reverzni potencijal curenja, V_T napon inicijacije impulsa, Δ_T faktor nagiba, w adaptacijska struja, a adaptacija ispod praga, τ_w vremenska konstanta adaptacije, a I_{syn} sinaptička struja. Impuls se dogodi kada V_m prijeđe V_{spike} , nakon čega se potencijal resetira na V_{reset} , a adaptacijska struja w se povećava za b , uz mogućnost uvođenja apsolutnog refraktornog razdoblja. Zbog računalne učinkovitosti i biološke vjernosti, AdEx se koristi kao kompromis između „integriraj i aktiviraj“ i biofizičkih modela [122].

U literaturi su opisane i manje zastupljene inačice „integriraj i aktiviraj“ modela koje proširuju osnovni formalizam dodatnim dinamičkim varijablama, nelinearnim strujama ili alternativnim mehanizmima generiranja impulsa [123], [124], [125], [126].

3.5.2 Izhikevich model neurona

Izhikevichev model neurona (IZH) je matematički model koji može reproducirati različite obrasce okidanja u kortikalnim neuronima [126], [95]. To postiže pomoću sustava jednadžbi koji bolje opisuju biološko ponašanje neurona, ali uz računalnu složenost sličnu „integriraj i aktiviraj“ modelima. Neuron je definiran dvjema diferencijalnim jednadžbama koje opisuju dinamiku potencijala membrane V i varijable oporavka u uz djelovanje ulazne struje I . Prva jednadžba prikazuje promjenu potencijala membrane koja dozvoljava modeliranje brze depolarizacije i repolarizacije.

$$\frac{dV}{dt} = 0.04v^2 + 5v + 140 - u + I \quad (3.12)$$

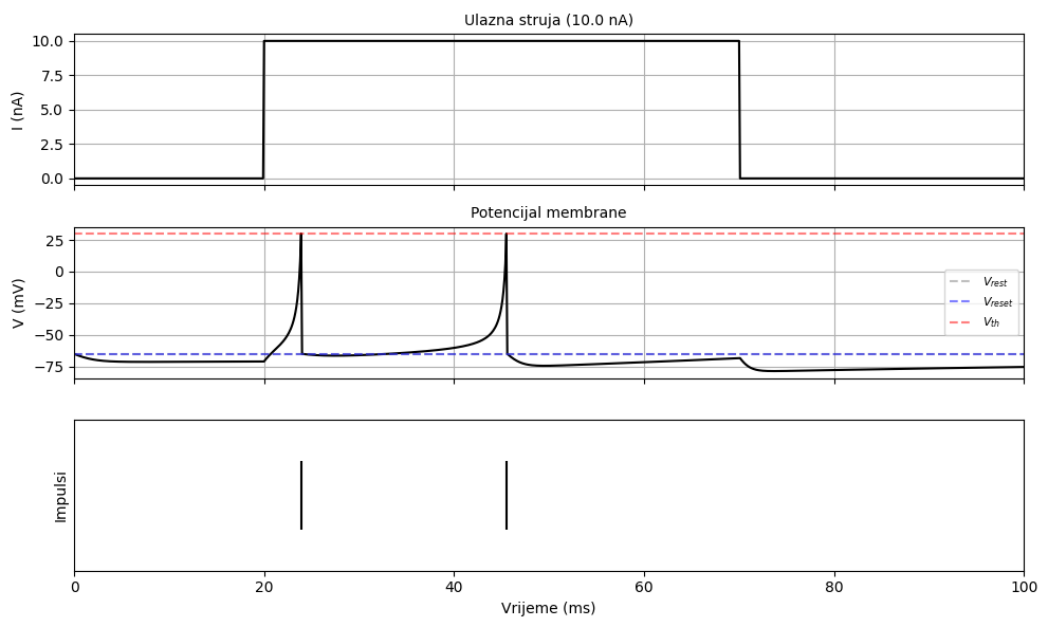
Druga jednadžba opisuje utjecaj promjene potencijala membrane na varijablu oporavka, čime se modelira mehanizam povratka neurona u stanje mirovanja nakon aktiviranja. Kada potencijal membrane dosegne definirani prag okidanja $V \geq V_{th}$ (obično $V_{th} = 30 \text{ mV}$), neuron generira impuls. Dinamika varijable oporavka je opisana izrazom:

$$\frac{du}{dt} = a(bV - u) \quad (3.13)$$

U trenutku generiranja impulsa, varijable V i u se resetiraju prema pravilima:

$$V \leftarrow c, \quad u \leftarrow u + d \quad (3.14)$$

čime se oponaša ponašanje bioloških neurona nakon akcijskog potencijala. Parametri a, b, c, d kontroliraju različite karakteristike ponašanja neurona. Parametar a određuje vrijeme oporavka u (obično između 0.02 i 0.1), a b definira osjetljivost varijable oporavka na promjene potencijala membrane (obično oko 0.2). Parametar c definira vrijednost resetiranja potencijala membrane nakon impulsa, a d određuje povećanje varijable oporavka nakon generiranja impulsa. Podešavanjem ovih vrijednosti, model može reproducirati širok raspon obrazaca neuronskih okidanja opaženih u biološkim sustavima [95].



Slika 3.7 Odgovor Izhikevichevog neuronskog modela na ulaznu struju amplitude $I = 10 \text{ nA}$. Prikazan je vremenski tijek potencijala membrane, pri čemu se tijekom primjene struje (20-70 ms) javlja ponavljano okidanje akcijskih potencijala, dok se nakon prestanka pobude neuron vraća u stanje mirovanja.

Slika 3.7 prikazuje promjene potencijala membrane kao odgovor na struju amplitude $I = 10 \text{ nA}$. Parametri modela postavljeni su na vrijednosti $a = 0.02, b = 0.2, c = -65 \text{ mV}, d = 8$. Početni potencijal neurona odgovara potencijalu mirovanja $V_{rest} = -65 \text{ mV}$, dok je potencijal okidanja postavljen na $V_{th} = 30 \text{ mV}$. Simulacija je provedena s vremenskim korakom $dt = 0.1$ tijekom 100 ms, pri čemu je ulazna struja primijenjena u intervalu od 20-70 ms. Tijekom djelovanja struje neuron generira niz akcijskih potencijala, a po njezinu prestanku se vraća u stanje mirovanja. Početno smanjenje potencijala membrane prije djelovanja ulazne struje posljedica neuravnoteženih početnih uvjeta. Varijabla u uvodi sporu repolarizaciju, što rezultira kratkotrajnom hiperpolarizacijom tj. stabiliziranjem sustava. Izhikevičev model u nekim situacijama može doseći nerealno visoke vrijednosti potencijala membrane ($>100 \text{ mV}$) zbog čega se maksimalni napon često ograničava na 30 mV što osigurava stabilnu simulaciju i biološki vjernije ponašanje [128], [127].

Postoji i manje korištena verzija modela, RF-IZH (*Resonate-and-Fire Izhikevich Neuron*), koja pojednostavljuje ponašanje neurona nakon impulsa, a zadržava istu dinamiku [123].

3.5.3 Hodgkin-Huxley model neurona

Hodgkin-Huxley (H-H) model pruža biofizički opis generiranja i širenja akcijskog potencijala u neuronu. Model je razvijen na temelju eksperimenata provedenih na divovskom aksonu lignje, čiji je veliki promjer omogućio precizno mjerenje ionskih struja. U modelu se razmatraju tokovi natrijevih i kalijevih iona te malih struja curenja, uglavnom sastavljeni od iona klora. Svaka struja opisana je vlastitom diferencijalnom jednačinom, gdje veličina toka ovisi o vodljivosti kanala (broju otvorenih kanala) i gradijentu (razlici između potencijala membrane i ravnotežnog potencijala iona) [110]. U izvornom obliku model opisuje tri vrste ionskih kanala, ali se može proširiti kako bi uključivao i druge tipove.

Na temelju eksperimenata, Hodgkin i Huxley pretpostavili su da se ukupna struja u membrani I_M sastoji od četiri komponente: natrijeve struje I_{Na} , kalijeve struje I_K , struje koju su nazvali strujom curenja I_L , koju nose većinom kloridni ioni, te struje I_E koju su sami ubrizgali [129].

$$I_M = I_{Na} + I_K + I_L + I_E \quad (3.15)$$

Natrijeva struja modelira se jednačinom:

$$I_{Na} = g_{Na} m^3 h (E_{Na} - V_M) \quad (3.16)$$

gdje g_{Na} predstavlja maksimalnu natrijevu vodljivost, odnosno najveću moguću struju ako su svi kanali otvoreni, a E_{Na} je reverzni potencijal natrija tj. napon pri kojem nema kretanja natrijevih iona kroz membranu. Ako je potencijal membrane V_M znatno niži od E_{Na} , natrijevi ioni ulaze u stanicu (depolarizacija). Umnožak varijabli otvaranja m i h određuje broj otvorenih natrijevih kanala. Upravljačka varijabla m raste brzo tijekom depolarizacije, povećavajući ulaz natrija. Istovremeno inaktivacijska varijabla h postupno opada i zatvara kanale natrija pa je ulaz natrija kratkotrajan, ali snažan [129].

Kalijeva struja opisana je jednačinom:

$$I_K = g_K n^4 (E_K - V_M) \quad (3.17)$$

gdje g_K označava maksimalnu kalijevu vodljivost, a E_K reverzni potencijal kalija. Kada se membrana depolarizira, aktivacijska varijabla n počinje rasti, ali sporije od m , što znači da se kalijevi kanali otvaraju sa zakašnjenjem. Ovo odgođeno povećanje vodljivosti proizvodi vanjsku struju (budući da je V_M obično pozitivniji od E_K) i pomaže u vraćanju membrane u stanje mirovanja nakon natrijevog skoka [129].

Struja curenja modelirana je kao mali, pasivni tok iona:

$$I_L = g_L(E_L - V_M) \quad (3.18)$$

gdje g_L predstavlja vodljivost curenja, a E_L reverzni potencijal kanala curenja. Ova struja je uvijek aktivna i stabilizira potencijal membrane u odsutnosti stimulacije [129].

Ukupna promjena potencijala membrane tijekom vremena određena je zbrojem ovih struja:

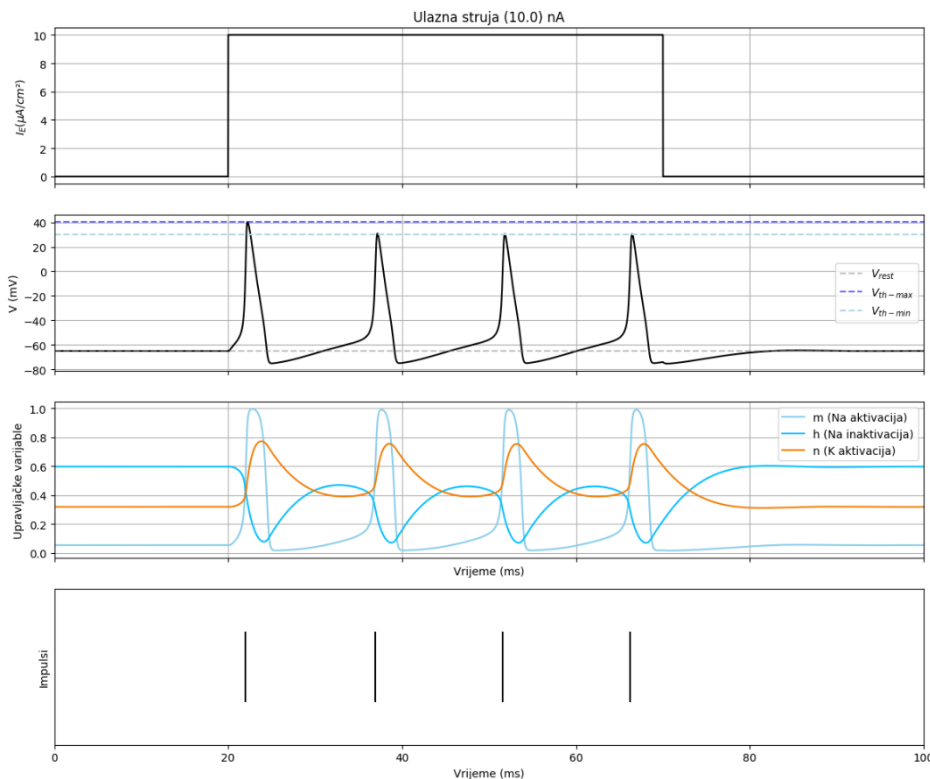
$$\frac{dV_M}{dt} = \frac{I_M}{C} \quad (3.19)$$

C označava kapacitet membrane (obično $1 \mu\text{F}/\text{cm}^2$) i definira oblik akcijskog potencijala. Kada dominira natrijeva struja, potencijal membrane brzo raste, a povećanje kalijeve struja dovodi do repolarizacije. Veći kapacitet usporava promjene napona, dok ga manji ubrzava.

Dinamika otvaranja i zatvaranja kanala kontrolirana je varijablama m, n i h , koje se kreću između 0 i 1 i predstavljaju vjerojatnost da su vrata kanala otvorena. Njihov tijek opisan je diferencijalnom jednačbom:

$$\frac{dx}{dt} = \alpha_x(1 - x) - \beta_x x \quad (3.20)$$

gdje x predstavlja m, n ili h , a α_x i β_x su naponski ovisne funkcije koje određuju brzinu prijelaza između otvorenog i zatvorenog stanja. Vrijednosti ovih parametara mogu varirati ovisno o eksperimentalnim podacima (Slika 3.8). Prvi akcijski potencijal doseže najveću amplitudu, dok su sljedeći impulsi nešto niži zbog djelomične inaktivacije Na kanala i povećane aktivacije K kanala.



Slika 3.8 Simulacija Hodgkin-Huxley modela neurona kao odgovor na vanjsku struju pobude. Struja amplitude $\frac{10 \mu\text{A}}{\text{cm}^2}$ je primijenjena u intervalu 10-70 ms, inducira akcijski potencijal praćen dinamikom varijabli m, h i n , koje opisuju aktivaciju i inaktivaciju natrijevih te aktivaciju kalijevih ionskih kanala. Simulacija koristi: $C_m = \frac{1 \mu\text{F}}{\text{cm}^2}$, $\bar{g}_{Na} = \frac{120 \text{ mS}}{\text{cm}^2}$, $\bar{g}_K = \frac{36 \text{ mS}}{\text{cm}^2}$, $\bar{g}_L = \frac{0.3 \text{ mS}}{\text{cm}^2}$, $E_{Na} = 50 \text{ mV}$, $E_K = -77 \text{ mV}$, $E_L = -54.4 \text{ mV}$ uz početni potencijal membrane $V = -65 \text{ mV}$.

4 Arhitektura impulsnih neuronskih mreža (SNN)

Impulsne neuronske mreže predstavljaju treću generaciju umjetnih neuronskih mreža, koje nastoje spojiti računalnu praktičnost umjetnih modela s načinom rada bioloških neurona. Za razliku od klasičnih neuronskih mreža, u kojima se informacija prenosi kontinuiranim vrijednostima, SNN-ovi koriste diskretne impulse kao osnovni komunikacijski mehanizam među neuronima. Informacija nije kodirana samo u jačini signala, već i u trenutku pojave impulsa, pa vrijeme postaje važan dio obrade podataka [130].

Radi jasnijeg sagledavanja sličnosti i razlika između umjetnih neuronskih mreža i impulsnih neuronskih mreža, Tablica 4.1 prikazuje usporedbu prema ključnim kategorijama.

Tablica 4.1 Usporedba impulsnih neuronskih mreža (SNN) i umjetnih neuronskih mreža (ANN).

Karakteristika		SNN	ANN
Kodiranje i reprezentacija podataka	Način kodiranja	prostorno-vremensko	kontinuirano (frekvencijsko)
	Reprezentacija podataka	diskretni impulsi	skalarnе vrijednosti
Simulacija	Model neurona	impulsni neuron	neuron s aktivacijskom funkcijom
	Način izračuna	diferencijalne jednačbe u vremenu	algebarski izračun (težinski zbroj + aktivacijska funkcija)
Učenje	Mehanizam učenja	STDP i njegove varijante	povratno širenje pogreške
Učinkovitost	Potrošnja energije	niska (događajno vođena)	visoka (kontinuirana aktivnost)

4.1 Topologija impulsnih neuronskih mreža

Topologija impulsnih neuronskih mreža odnosi se na način organizacije neurona i sinaptičkih veza unutar mreže koja slijedi koncepte poznate iz umjetnih neuronskih mreža, ali uz dodatne zahtjeve vezane uz vremensku dinamiku. Kao što je prikazano na Slika 4.1 i sažeto u Tablica 4.2, impulsna neuronska mreža može se interpretirati kroz više bioloških analogija, ovisno o kriteriju klasifikacije neurona definiranom u prethodnom poglavlju.

Tablica 4.2 Biološke analogije umjetnih neurona prema različitim kriterijima podjele.

	Ulazni neuron	Skriveni neuron	Izlazni neuron
Kriterij podjele			
Funkcionalna podjela	senzorni	interneuron	motorni
Morfološka podjela	pseudounipolarni / bipolarni	pseudounipolarni / bipolarni / multipolarni	multipolarni
Sinaptička uloga	presinaptički	presinaptički / postsinaptički	postsinaptički
Učinak na signal	ekscitatorni / inhibitorni	ekscitatorni / inhibitorni	ekscitatorni / inhibitorni

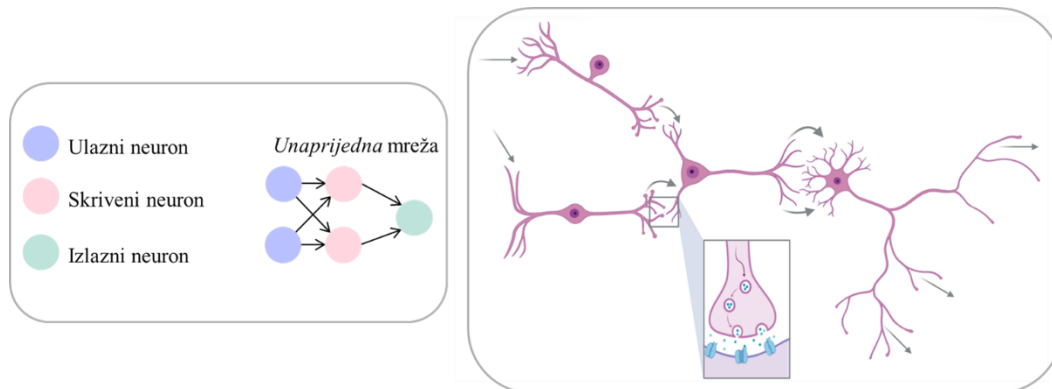
Najjednostavniji oblik mreže čine unaprijed povezane (engl. *feedforward*) strukture, u kojima se informacija širi jednosmjerno, od ulaznog prema izlaznom sloju [5], [105]. Iako ove arhitekture nemaju povratne veze, dinamika impulsnih neurona omogućuje im da integriraju informacije kroz vrijeme i uzmu ga u obzir kod predikcije uzoraka.

Složeniji oblici, poput rekurentnih impulsnih mreža, koriste povratne sinaptičke veze za kruženje informacija unutar sustava. Ovakva strukturna rekurentnost zajedno s ponašanjem

potencijala membrane, omogućuje mreži da prirodno implementira oblik kratkoročne memorije. Za razliku od klasičnih rekurentnih mreža koje zahtijevaju memorijske ćelije (poput LSTM jedinica), SNN-ovi zadržavaju tragove prošlih događaja unutrašnjim stanjima neurona [125].

Na višim razinama apstrakcije, SNN-ovi se organiziraju u hijerarhijske višeslojne arhitekture i konvolucijske impulsne mreže [131], [132], [133]. U takvim strukturama svaki sloj postupno izdvaja značajke na različitim razinama apstrakcije, dok se vremenska komponenta signala prenosi kroz cijelu hijerarhiju. Konvolucijske impulsne mreže prepoznaju vizualne obrasce analizirajući male grupe susjednih neurona. One su iznimno učinkovite u računalnom vidu, osobito kada obrađuju podatke iz neuromorfni senzora koji, poput ljudskog oka, šalju informacije samo o promjenama u sceni [134].

Osim pravilnih slojevitih struktura, u SNN-ovima se koriste i rijetko povezane topologije, inspirirane biološkim neuronskim mrežama. U takvim mrežama svaki neuron nije povezan sa svima ostalima, već samo s manjim brojem neurona, što dovodi do rjeđe i nepravilnije strukture. Dok se kod klasičnih neuronskih mreža koriste gusto povezane i strogo definirane arhitekture, rijetka povezanost u SNN-u smanjuje računalnu složenost i potrošnju energije a raznolikost neurona omogućuje mreži da bolje prilagodi svoje ponašanje različitim zadacima [4].



Slika 4.1 Komparativni prikaz arhitekture neuronskih mreža. Lijevo: Klasična umjetna unaprijedna neuronska mreža sa slojevima ulaznih, skrivenih i izlaznih neurona. Desno: Shematski prikaz impulsne neuronske mreže prikazan kroz morfološku analogiju biološkog neurona.

4.1.1 Koncept rijetkosti i biološka adaptacija

Važno obilježje SNN-ova je rijetkost (engl. *sparsity*) koja je inspirirana biološkim principom energetske učinkovitosti. Za razliku od ANN modela gdje svaki neuron, u svakom sloju, izračunava izlaznu vrijednost pri svakom prolazu kroz mrežu, neuroni u SNN-u većinu vremena ostaju neaktivni te emitiraju impulse samo kada akumulirani ulaz premaši prag pobude, što smanjuje računalno opterećenje [135]. Ovakvo ponašanje se može promatrati na dvije razine. Prostorna rijetkost znači da je u svakom trenutku aktivan samo manji broj neurona. Primjer toga je ljudska mrežnica koja reagira na promjene. Kod statične slike većina neurona ostaje neaktivna, a aktiviraju se tek pri promjeni svjetline [136]. S druge strane, vremenska rijetkost odnosi se na to da neuroni ne reagiraju stalno na isti podražaj već se njihova aktivnost s vremenom smanjuje ako se podražaj ne mijenja. Primjer je osjećaj odjeće na tijelu, koji ubrzo nakon odijevanja prestajemo primjećivati. Ovakav način rada ne utječe samo na potrošnju energije, već i na način obrade informacija zadržava.

Dinamika koja započinje na razini neurona prerasta u kompleksno kolektivno ponašanje. Sinkronizacija impulsa omogućuje pojačavanje bitnih uzoraka, dok asinkrona aktivnost pridonosi stabilnosti i robusnosti sustava. Ovi fenomeni nemaju ekvivalent u ANN-ovima, jer

zahtijevaju posebne metode implementacije. Budući da računala rade u diskretnim koracima, kontinuirano ponašanje neurona mora se aproksimirati vremenskim korakom (Δt). Prevelik korak smanjuje preciznost, dok premali korak značajno povećava računalni trošak.

Važno je istaknuti da topologija i vremenska dinamika ne određuju samo tok informacija, već izravno uvjetuju i mogućnosti učenja. Određene strukture prirodno pogoduju lokalnim biološkim pravilima, dok druge zahtijevaju globalne metode optimizacije, zbog čega se dizajn topologije SNN-a često razmatra u uskoj povezanosti s odabranom paradigmom učenja.

4.2 Učenje u SNN-ovima

U impulsnim neuronskim mrežama sinapse se ne promatraju samo kao kanali prijenosa informacija, već su zadužene za proces učenja i memoriju. Svaka je veza definirana sinaptičkom težinom koja određuje stupanj utjecaja presinaptičkog neurona na postsinaptički. Dobiveni postsinaptički odgovor na dolazne impulse pokazuje važnost pojedinih neurona. Kako bi se vjerne replicirale biološke funkcije depolarizacije, repolarizacije i hiperpolarizacije, SNN-ovi koriste sustav pozitivnih i negativnih težina koji simuliraju ekscitacijske (pobuđujuće) i inhibicijske (prigušujuće) učinke. Dok ekscitacija teži aktivaciji neurona, inhibicijski procesi smanjuju njegovu aktivnosti. Inhibicijski neuroni omogućuju stabilnost sustava i sprječavaju nekontrolirano širenje impulsa. Posebno su značajni mehanizmi kompeticije i lateralne inhibicije. Jedan od najčešće korištenih primjera takvog ponašanja jest princip „pobjednik uzima sve” (engl. *Winner-Take-All*, WTA), koji dopušta isključivo najaktivnijim neuronima unutar sloja da generiraju impulse prema sljedećoj razini, dok se manje aktivni neuroni potiskuju. Drugi važan mehanizam inhibicije koji se nadovezuje na prethodno opisane kompeticijske procese je lateralna inhibicija. Ovaj fenomen je prisutan u biološkim neuronskim strukturama, pri kojem aktivacija jednog neurona izravno dovodi do smanjenja ekscitacije njegovih susjednih jedinica. Za razliku od mehanizama kompeticije koji su usmjereni na globalnu dominaciju pojedinih neurona, lateralna inhibicija djeluje kao lokalizirani proces. Njezin je primarni cilj potiskivanje obližnjih neurona kako bi se poboljšala selektivnost i jasnoća odgovora mreže na vanjske podražaje [89], [137], [138].

Dok prethodno opisani inhibicijski i kompeticijski mehanizmi reguliraju trenutnu aktivnost neuronske mreže, dugoročna prilagodba i sposobnost učenja u SNN-ovima temelje se na mehanizmima koji omogućuju vremensku promjenjivost sinapsi. Sinaptička plastičnost omogućuje mreži pohranu informacija i prilagodbu ponašanja na temelju prethodnih podražaja, pri čemu se jačina veza između neurona postupno povećava ili smanjuje ovisno o njihovoj međusobnoj aktivnosti. Premda je krajnji cilj učenja optimizacija težina, isti kao u ANN-ovima, sama izvedba unutar SNN-a je specifična zbog asinkrone prirode impulsa [89]. Trenutni pristupi treniranju impulsnih mreža mogu se podijeliti u tri glavne kategorije:

- 1) **Učenje pod nadzorom:** Oslanja se na prilagođene algoritme gradijentnog spusta i povratne propagacije pogreške kroz vrijeme (engl. *backpropagation*). Iako precizna, ova metoda se suočava s problemom nediferencijabilnosti impulsa.
- 2) **Učenje bez nadzora:** Temelji se na lokalnim pravilima, prvenstveno na plastičnosti ovisnoj o vremenu ispaljivanja impulsa (STDP). Iako je biološki najuvjerljiviji, ovaj pristup često rezultira nižom točnošću u usporedbi s metodama učenja pod nadzorom.
- 3) **Učenje potkrepljivanjem:** Koristi vanjske signale nagrade ili kazne kako bi se modulirala plastičnost i usmjerilo ponašanje mreže.

Zbog složenosti treniranja, jedna od trenutno najuspješnijih strategija u praksi je konverzija modela. Ovaj pristup podrazumijeva obuku ANN-a koji se zatim prevodi u SNN arhitekturu. Na taj se način koriste prednosti razvijenih algoritama dubokog učenja, uz zadržavanje energetske učinkovitosti impulsnih sustava [139].

Bez obzira na pristup treniranju, rad neuronske mreže temelji se na prilagodbi veza između neurona. Neuroni i i j povezni su težinom w_{ij} čija je svrha prilagoditi ponašanje mreže datom zadatku. Proces kojim se te težine mijenjaju naziva se učenje, a skup pravila prema kojima se promjene provode naziva se pravilo učenja. Proces učenja se obično odvija primjenom metode povratnog širenja greške. Ovaj algoritam izračunava gradijente, odnosno derivacije funkcije troška u odnosu na parametre mreže, koji opisuju smjer i brzinu promjene vrijednosti pogreške. Na temelju signala pogreške, gradijenti se propagiraju unatrag kroz sve slojeve neuronske mreže. Za ažuriranje težina najčešće se koristi algoritam gradijentnog spusta. Kako bi povratno širenje pogreške bilo primjenjivo, aktivacijska funkcija mora biti diferencijabilna, ali u SNN-ovima neuron generira impuls kao diskretan događaj, koji se može opisati *step*-funkcijom koja prelazi iz vrijednosti 0 na 1. Derivacija takve funkcije u trenutku okidanja je beskonačna, dok je u svim ostalim točkama jednaka nuli. Zbog toga klasične metode optimizacije temeljene na gradijentima nailaze na ograničenja u SNN-ovima.

Surogatni gradijent je tehnika koja rješava problem nediferencijabilnih aktivacijskih funkcija tako da njihovu derivaciju aproksimira pomoću diferencijabilnih funkcija sličnog ponašanja. Neka aktivacijska funkcija $f(x)$ generira impuls kada ulaz x prijeđe određeni prag. Budući da je ta funkcija diskontinuirana i nediferencijabilna, nije moguće izravno izračunati gradijent niti primijeniti standardne metode učenja temeljene na propagaciji pogreške unatrag. Tijekom prolaza unaprijed, neuron se ponaša biološki realistično i generira diskretne impulse kada potencijal membrane prijeđe prag aktivacije. Tijekom propagacije pogreške unatrag, derivacija *step*-funkcije aproksimira se glatkom i kontinuiranom funkcijom. Takva funkcija naziva se surogatna funkcija $g(x)$ koja služi kao diferencijabilna aproksimacija ponašanja izvorne funkcije u okolini praga aktivacije [140]. Učenje se formulira kao problem minimizacije funkcije gubitka L , a često se koristi kvadratna funkcija pogreške:

$$L = \frac{1}{2}(y - g(x))^2 \quad (4.1)$$

gdje je y željeni izlaz, a $g(x)$ izlaz surogatne funkcije. Kako bi se omogućilo ažuriranje parametara mreže, potrebno je izračunati gradijent funkcije gubitka u odnosu na ulaz. Funkcija gubitka ovisi o x , putem izlaza $g(x)$, gradijent se računa primjenom pravila lanca:

$$\frac{\partial L}{\partial x} = \frac{\partial L}{\partial g(x)} * \frac{\partial g(x)}{\partial x} \quad (4.2)$$

Deriviranjem kvadratne funkcije pogreške po $g(x)$ dobiva se:

$$\frac{\partial L}{\partial g(x)} = g(x) - y \quad (4.3)$$

pa ukupni izraz za gradijent poprima oblik:

$$\frac{\partial L}{\partial x} = (g(x) - y) * \frac{\partial g(x)}{\partial x} \quad (4.4)$$

Kako bi se omogućilo računanje derivacije, surogatna funkcija može se definirati kao kompozicija nediferencijabilne funkcije $f(x)$ i diferencijabilna funkcije $\sigma(x)$, najčešće logističke funkcije:

$$g(x) = \sigma(f(x)) \quad (4.5)$$

Primjenom pravila lanca dobiva se:

$$\frac{dg}{df} = \sigma(f(x)) * (1 - \sigma(f(x))) \quad (4.6)$$

$$\frac{\partial g(x)}{\partial x} = \frac{\partial \sigma}{\partial f} * \frac{\partial f}{\partial x} \quad (4.7)$$

U slučaju kada je $f(x)$ step-funkcija s pragom θ , njezina derivacija može se formalno opisati Diracovom delta funkcijom koja matematički modelira trenutačni događaj da poprima vrijednost nula u svim trenucima osim u točki pojave impulsa, kada uzrokuje diskretno povećanje traga:

$$\frac{\partial f}{\partial x} = \delta(x - \theta) \quad (4.8)$$

Ako sve saberemo zajedno, dobivamo izraz za surogatni gradijent koji omogućuje prijenos signala pogreške kroz impulsnu aktivacijsku funkciju tijekom procesa učenja, unatoč njezinoj nediferencijabilnosti:

$$\frac{\partial g}{\partial x} = \sigma(f(x)) * (1 - \sigma(f(x))) * \delta(x - \theta) \quad (4.9)$$

4.2.1 Nadzirano učenje

Nadzirano učenje predstavlja paradigmu treniranja modela gdje je svakom ulaznom podatku pridružena odgovarajuća ciljana izlazna vrijednost. Ta vrijednost služi kao referentni signal na temelju kojeg se izračunava pogreška i prilagođavaju parametri modela. Proces nadziranog učenja u SNN-ovima može se podijeliti u nekoliko faza. U prvoj fazi ulazni se podaci kodiraju u nizove impulse $s_i^n(t) \in S_i$ primjenom odgovarajuće metode kodiranja. Kodirani podaci se dovode na ulaz mreže sa odabranim modelom neurona i sinapsi. Kao rezultat se dobivaju stvarni izlazni impulsi nizovi $s_o^m(t) \in S_o$. Na temelju željenih izlaznih impulsi nizova $s_d^m(t) \in S_d$ izračunava se vrijednost funkcije pogreške $E(S_o, S_d)$, nakon čega se sinaptičke težine ažuriraju prema pravilu $W \leftarrow W + \Delta W$. Proces učenja iterativno se ponavlja sve dok pogreška ne padne ispod unaprijed zadanog praga ili dok se ne dosegne maksimalan broj epoha učenja. Nakon završetka učenja izlazni impulsi nizovi se dekodiraju.

Postojeća pravila nadziranog učenja u SNN-ovima se mogu klasificirati prema tri glavna kriterija; struktura mreže definira broj slojeva za koje je pravilo razvijeno, matematička osnova su teorijski principi na kojima se pravilo temelji i utječe na razinu biološke vjerodostojnosti. Posljednji je broj impulsa koje neuron može generirati čime se određuje količina informacija koja se može prenijeti kroz mrežu [141].

SpikeProp [142] je jedan od najranijih algoritama koji primjenjuje koncepte povratnog širenja pogreške u vremenskoj domeni. Izveden je prema principima sličnima klasičnoj metodi povratnog širenja pogreške koja se koristi u ANN-ovima. Algoritam prilagođava sinaptičke težine na temelju razlike između stvarnih i željenih vremena okidanja impulsa neurona. Ažuriranje težina opisano je izrazom:

$$\Delta w_{ij} = -\eta \sum_d \frac{\partial E}{\partial t_i^d} \frac{\partial t_i^d}{\partial w_{ij}} \quad (4.10)$$

gdje je η stopa učenja, E mjeri neusklađenost između stvarnih i željenih vremena impulsa t_i^d , a $\partial t_i^d / \partial w_{ij}$ je osjetljivost vremena impulsa. Funkcija pogreške koja se koristi je metoda srednje kvadratne pogreške koja se propagira unatrag kroz skrivene slojeve mreže te se minimizira optimizacijom sinaptičkih težina. Predloženo je nekoliko varijante ove metode [143], [144] s

ciljem poboljšanja brzine konvergencije i točnosti klasifikacije ali unatoč poboljšanjima sva pravila dijele isto ograničenje, izlazni neuroni mogu generirati najviše jedan impuls čime je ograničena količina informacije koju neuron može prenijeti.

SuperSpike [145] proširuje gornji algoritam uvođenjem surogatnih gradijenata i sinaptičkih tragova podobnosti (engl. *eligibility traces*) koji zadržavaju informaciju o prošloj sinaptičkoj aktivnosti, čime se omogućuje učenje višeslojnih SNN-ova. Pravilo ažuriranja težina može se zapisati u obliku:

$$\Delta w_{ij} = \eta \int_{t_b}^{t_{b+1}} e_i(t) [\alpha * (\sigma'(U_i(t)) \cdot (\epsilon * S_j)(t))] dt \quad (4.11)$$

gdje $e_i(t)$ predstavlja signal pogreške dobiven primjenom surogatnog gradijenta impulsne funkcije $\sigma'(U_i(t))$ i presinaptičkim tragom $(\epsilon * S_j)(t)$. Iako omogućuje treniranje skrivenih slojeva i ne ograničava neuron na okidanje jednog impulsa, još uvijek zahtijeva aproksimaciju gradijenta jer impuls signal nema derivaciju i pokazuje probleme stabilnosti i skalabilnosti kod dubokih mreža.

SLAYER (*Spike Layer Error Reassignment in Time*) [146] je dizajniran da riješi problem diferencijabilnosti i vremenske raspodjele pogreške u SNN-ovima. U klasičnom algoritmu povratnog širenja pogreške, pogreška se propagira kroz slojeve, ali u SNN modelima s diskretnim događajima impulsa teško je izračunati gradijente u vremenu. SLAYER uvodi vremensko raspoređivanje pogreške što znači da pogreška nije vezana samo uz trenutnu aktivaciju, nego se širi unatrag kroz vrijeme i impulsne događaje. Općeniti oblik ažuriranja sinaptičkih težina predstavlja surogatni gradijent impulsnog niza $\partial s(t)/\partial w$ u odnosu na sinaptičku težinu i dan je izrazom:

$$\Delta w_{ij} = -\eta \sum_t \frac{\partial E}{\partial s(t)} \frac{\partial s(t)}{\partial w} \quad (4.12)$$

EventProp [147] koristi drugačiji pristup izračunu gradijenata, pri čemu se diskontinuiteti derivacija u trenucima okidanja impulsa eksplicitno obrađuju primjenom adjungiranog sustava jednačbi, bez korištenja surogatnih gradijenata. Funkcija gubitka definirana je kao:

$$L = l_p(t_{post}) + \int_0^T l_V(V(t), t) dt \quad (4.13)$$

a njegov gradijent u odnosu na sinapsu w_{ij} je:

$$\frac{dL}{dw_{ij}} = -\tau_{syn} \sum_{impuls\ od\ i} (\lambda_I)_j \quad (4.14)$$

gdje je $(\lambda_I)_j$ adjungirana vrijednost sinaptičke struje, a τ_{syn} sinaptička konstanta. Ovo je formalno istovjetno metodi povratnog širenja pogreške u tradicionalnim mrežama, ali razvijeno za rad s impulsima. Iako ima prednost memorijske efikasnosti za događajno vođene simulacije jer zahtjeva poznavanje stanja u trenucima impulsa, ne skalira se dobro na duboke arhitekture.

4.2.2 Nenadzirano učenje

Nenadzirano učenje ne koristi unaprijed definirane ciljne izlaze već je cilj otkriti skrivenu strukturu, obrasce ili statističke pravilnosti unutar samih podataka. U kontekstu SNN-ova, takvo se učenje često temelji na lokalnim pravilima prilagodbe sinaptičkih težina koja iskorištavaju vremenske odnose između impulsa neurona.

Među brojnim postojećim pravilima učenja, posebno se ističe skupina lokalnih pravila kod kojih su promjene sinaptičkih veza izravno uvjetovane zajedničkom aktivnošću presinaptičkih i postsinaptičkih neurona. Ovakav pristup proizlazi iz Hebbovog učenja, prema kojem ponovljena istodobna aktivacija povezanih neurona dovodi do jačanja njihove međusobne veze [148], [149]. Ovo se pravilo u literaturi često sažima tvrdnjom da veze između neurona jačaju kada oni istodobno izbijaju impulse („*neurons that fire together, wire together*”). Formalni opis Hebbova pravila može se izraziti dinamikom promjene sinaptičkih težina w :

$$\tau_w \frac{dw}{dt} = vu \quad (4.15)$$

U ovoj jednadžbi w označava sinaptičku težinu tj. matricu težina, v izlaznu brzinu okidanja, a u predstavlja sinaptički ulaz. Parametar τ_w je vremenska konstanta koja određuje brzinu promjene sinaptičkih težina. Iako ovaj pristup pruža intuitivan uvid u osnovni mehanizam učenja, on ima značajna ograničenja. Prije svega, model ne obuhvaća inhibicijske sinapse, već je primjenjiv isključivo na ekscitacijske veze, a bez dodatnih stabilizacijskih mehanizama, vrijednosti težina mogu rasti bez ograničenja, što dovodi do nestabilnog ponašanja mreže [89], [150].

Jedan od biološki preciznijih oblika Hebbovog učenja je plastičnost ovisna o vremenskom razmaku izbijanja neurona (engl. *Spike-Timing-Dependent Plasticity*, STDP) [151], [149]. Za razliku od klasičnog Hebbovog pravila koje naglašava korelaciju aktivnosti neurona, STDP uzima u obzir točno vrijeme pojave presinaptičkih i postsinaptičkih impulsa pa promjena sinaptičke težine ovisi o relativnom vremenskom razmaku između njih. Promjena težina temelji se na lokalnoj vremenskoj dinamici impulsa, bez potrebe za vanjskim signalom pogreške ili globalnim nadzorom. Ako presinaptički neuron generira impuls prije postsinaptičkog, sinaptička veza se pojačava. Taj proces naziva se dugotrajna potencijacija (engl. *Long-Term Potentiation*, LTP). U tom slučaju smatra se da je presinaptička aktivnost uzročno doprinijela aktivaciji postsinaptičkog neurona, što odgovara Hebbovom pravilu. Suprotno tome, ako presinaptički impuls nastupi nakon postsinaptičkog, sinaptička veza slabi kroz proces poznat kao dugotrajna depresija (engl. *Long-Term Depression*, LTD). Ovaj oblik promjene često se naziva anti-Hebbovom plastičnošću jer presinaptička aktivnost nije doprinijela aktivaciji postsinaptičkog neurona [151], [152]. Matematički, promjena sinaptičke težine može se izraziti kao:

$$\Delta w = \begin{cases} A^+ e^{\left(\frac{t_{pre} - t_{post}}{\tau^+}\right)}, & t_{pre} < t_{post} \\ A^- e^{\left(-\frac{t_{pre} - t_{post}}{\tau^-}\right)}, & t_{pre} > t_{post} \end{cases} \quad (4.16)$$

gdje je Δw promjena sinaptičke težine, t_{pre} i t_{post} vrijeme presinaptičkog i postsinaptičkog impulsa ($\Delta t = t_{pre} - t_{post}$ njihov vremenski razmak). A^+ i A^- su maksimalne amplitude potencijacije i depresije, a utjecaj vremena na opadanje impulsa određuju vremenske konstante τ^+ i τ^- . Kada je $\Delta t > 0$, odnosno kada presinaptički impuls prethodi postsinaptičkom, dolazi do jačanja veze (LTP), a kada je $\Delta t < 0$, veza slabi (LTD) [153], [151].

Računalna implementacija STDP pravila bi zahtijevalo pohranu i usporedbu točnih vremena svih presinaptičkih i postsinaptičkih impulsa pa se u praksi koristi aproksimacija temeljena na tzv. tragovima aktivnosti (engl. *traces*). Ova formulacija poznata je kao online STDP pravilo, jer omogućuje lokalno, inkrementalno i vremenski kontinuirano ažuriranje sinaptičkih težina čime se postiže manja računalna složenost uz zadržavanje biološke uvjerljivosti [151], [154]. U ovom pristupu svaki presinaptički neuron j ostavlja privremeni trag $x_j(t)$, koji predstavlja eksponencijalno opadajuću funkciju vremena i opisuje „memoriju” nedavne presinaptičke

aktivnosti. Analogno tome, postsinaptički neuron generira trag $y(t)$ koji sažima njegovu aktivnost. Dinamika ovih tragova opisana je diferencijalnim jednadžbama [151]:

$$\tau^+ \frac{dx_j}{dt} = -x_j + a^+(x_j) \sum_f \delta(t - t_j^f) \quad (4.17)$$

$$\tau^- \frac{dy}{dt} = -y + a^-(y) \sum_n \delta(t - t^n) \quad (4.18)$$

gdje t_j^f označava vrijeme f -tog presinaptičkog impulsa neurona j , t^n vrijeme n -tog postsinaptičkog impulsa, a^+ i a^- su inkrementi kojima se trag povećava u trenutku pojave impulsa, a τ^+ i τ^- su vremenske konstante koje određuju brzinu eksponencijalnog opadanja tragova. U odsutnosti novih impulsa (x_j, y) tragovi eksponencijalno opadaju prema nuli.

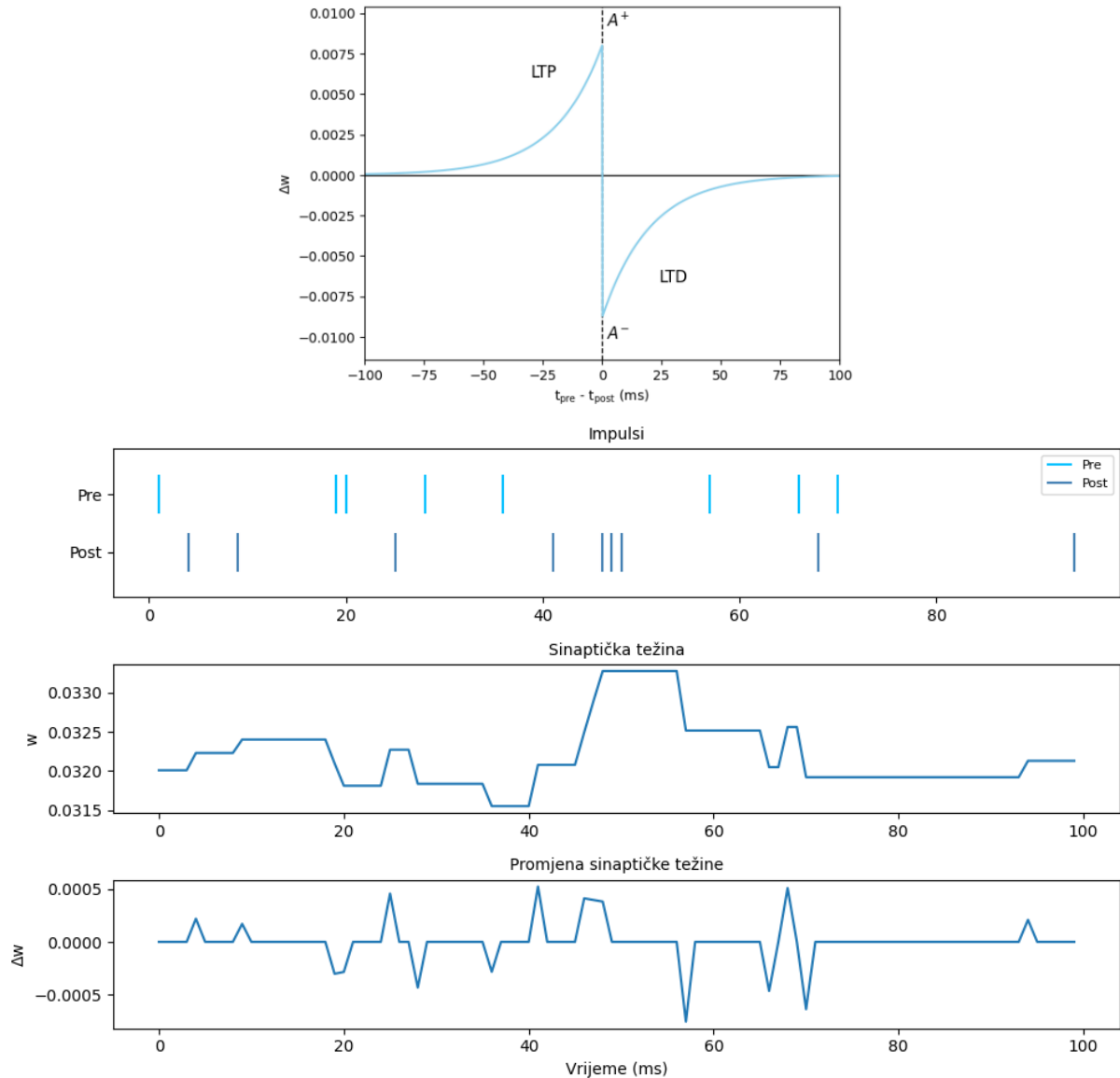
$$\frac{dw_j}{dt} = A^+(w_j)x_j(t) \sum_n \delta(t - t^n) - A^-(w_j)y(t) \sum_f \delta(t - t_j^f) \quad (4.19)$$

Promjena sinaptičke težine w_j se definira lokalno u vremenu gdje su $A^+(w_j)$ i $A^-(w_j)$ faktori potencijacije i depresije, koji u općenitom slučaju mogu ovisiti o trenutnoj vrijednosti težine (npr. radi sprječavanja njezina nekontroliranog rasta ili zasićenja). Iz ove formulacije proizlazi da sinaptička težina raste proporcionalno presinaptičkom tragu $x_j(t)$ u trenutku postsinaptičkog impulsa (kada je aktivna suma $\sum_n \delta(t - t^n)$). Budući da trag kodira nedavnu presinaptičku aktivnost, dolazi do dugotrajne potencijacije (LTP) ako je presinaptički neuron bio aktivan neposredno prije postsinaptičkog impulsa. U trenutku presinaptičkog impulsa (kada je aktivna suma $\sum_f \delta(t - t_j^f)$), težina se smanjuje proporcionalno postsinaptičkom tragu $y(t)$ ako je postsinaptički neuron proizveo impuls, dolazi do dugotrajne depresije (LTD). Promjena težine događa se isključivo u diskretnim trenucima pojave impulsa, dok između impulsa vrijedi $\frac{dw_j}{dt} = 0$. Integracijom ove jednadžbe kroz vrijeme dobiva se funkcionalni oblik plastičnosti koji aproksimira izvorno STDP pravilo, ali uz znatno jednostavniju implementaciju. Važno je napomenuti da promjene sinaptičke težine (Δw) nije neposredan odgovor na impuls presinaptičkog neurona. Umjesto toga, STDP mijenja težinu ovisno o vremenskom razmaku između impulsa pre- i postsinaptičkog neurona. U praksi, kada presinaptički neuron emitira impuls, Δw ostaje nula dok postsinaptički neuron još nije emitirao impuls. Tek kada dođe do impulsa postsinaptičkog neurona, prethodno pohranjeni trag uzrokuje pojačanje ili slabljenje sinapse stoga je zakašnjenje vidljivo na Slika 4.2 očekivano ponašanje STDP-a.

Iako je STDP temeljni model vremenski ovisne sinaptičke plastičnosti, brojna su istraživanja predložila njegova proširenja s ciljem povećanja biološke vjerodostojnosti, stabilnosti učenja te ograničenja fiksne dinamike. Za modeliranje složenijih bioloških interakcija, razvijen je trostruki STDP [155]. Za razliku od klasičnog parnog modela, ovaj pristup analizira sekvence od tri impulsa (npr. *pre-post-pre*) što omogućuje modeliranje plastičnosti ovisne o frekvenciji impulsa (engl. *rate-dependent plasticity*) i postiže veća biološka točnost u usporedbi s parnim STDP-om.

Multiplikativni STDP mijenja težinu sinapse ovisno o njezinoj trenutačnoj vrijednosti. Promjene se smanjuju kada je težina već velika, a povećavaju kada je mala. Time se prirodno ograničava rast težina jer jake sinapse teže daljnjem jačanju, a vrlo slabe sinapse se ne gase u potpunosti [156]. Slično, adaptivni STDP (aSTDP) uvodi dinamičku prilagodbu parametara tijekom procesa treniranja, omogućujući regulaciju vremenskih prozora i amplituda za dugotrajno jačanje (LTP) ili slabljenje (LTD) sinapsi [157]. Osim varijacija STDP-a, u literaturi

se ističu i drugi modeli koji adresiraju specifične aspekte sinaptičke dinamike. Ojaovo pravilo proširuje Hebbovo učenje tako što ograničava rast težina, čime se sprječava njihovo nekontrolirano povećavanje i održava stabilnost mreže [158]. Sličan pristup ima i Bienenstock-Cooper-Munro (BCM) pravilo [159] koje uvodi prag koji se mijenja ovisno o prosječnoj aktivnosti neurona, pa se tako učenje prilagođava stanju mreže. Osim dugoročne prilagodbe težina, postoje i modeli kratkotrajne plastičnosti (engl. *Short-Term Plasticity*, STP) [160] koji opisuju kratke i prolazne promjene u snazi sinapsi.



Slika 4.2 STDP krivulja koja prikazuje promjenu sinaptičke težine (Δw) u ovisnosti o vremenskoj razlici između presinaptičkog i postsinaptičkog impulsa ($t_{pre} - t_{post}$). Parametri modela iznose $A^+ = 0.008$, $A^- = 0.0088$ te $\tau^+ = \tau^- = 20$ ms. Za $t_{pre} - t_{post} < 0$ dolazi do dugoročnog pojačanja sinapse (LTP), dok za $t_{pre} - t_{post} > 0$ dolazi do dugoročnog slabljenja (LTD). Promjena težine eksponencijalno opada s vremenskom konstantom τ .

Unatoč navedenim proširenjima, opisane varijante STDP-a ostaju lokalna pravila učenja temeljena na vremenskoj korelaciji impulsa i same po sebi ne optimiziraju funkciju pogreške. To ograničava njihovu primjenjivost u dubokim arhitekturama SNN-ova gdje je za učinkovito treniranje često potrebna globalna informacija o pogrešci modela. U tom se kontekstu uvode pristupi temeljeni na potkrepljenju koji omogućuju integraciju globalnog signala nagrade u proces sinaptičke prilagodbe.

4.2.3 Učenje potkrepljivanjem

Učenje u biološkim sustavima često je vođeno sustavom nagrada; životinje uče povezivati specifične akcije s pozitivnim ishodima, što dovodi do selektivnog jačanja onih veza koje su pridonijele uspjehu. Ovakvo oblikovanje na temelju povratne informacije važno je u brzim, sekvencijalnim zadacima gdje povratna informacija o uspješnosti akcije dolazi tek nakon što je cijela sekvenca završena. Standardni STDP modeli ne mogu u potpunosti objasniti ovaj fenomen jer im nedostaje mehanizam za razlikovanje uspješnih od neuspješnih akcija. Ovaj nedostatak prevladava se uvođenjem trećeg signala, najčešće neuromodulatora⁴, koji proširuje Hebbovo pravilo [152].

Za razliku od STDP-a, STDP moduliran nagradom (engl. *Reward-Modulated STDP*, R-STDP) ne modificira sinapse isključivo na temelju vremena okidanja impulsa, već uzima u obzir je li ishod neuronske aktivnosti bio koristan za sustav. Matematički opis R-STDP-a obično se temelji na dvije komponente: izračunu traga podobnosti i pravilu za ažuriranje sinaptičkih težina [105], [161]. Trag podobnosti (e_{ij}) djeluje kao svojevrsna kratkoročna memorija na razini svake sinapse jer bilježi nedavne vremenske interakcije između presinaptičkog neurona j i postsinaptičkog neurona i . Dinamika ovog traga opisuje se sljedećom diferencijalnom jednačinom:

$$\tau_e \frac{de_{ij}}{dt} = -e_{ij}(t) + STDP(pre_j, post_i) \quad (4.20)$$

Vrijednost traga raste sa svakim ispravnim redosljedom impulsa (prema standardnom STDP pravilu), dok u nedostatku aktivnosti eksponencijalno opada s vremenskom konstantom τ_e . Primarna svrha traga podobnosti je „označavanje” nedavno aktivnih sinapsi, čime one postaju podložne promjenama u trenutku kada sustav primi signal nagrade. Ovaj je mehanizam ključan kod odgođenih nagrada jer omogućuje mreži da zadrži informaciju o tome koje su sinapse sudjelovali u aktivnosti, bez trenutne promjene njihove snage [105], [152].

Stvarna promjena sinaptičke težine w_{ij} se događa pod utjecajem signala nagrade $R(t)$ prema pravilu:

$$\frac{dw_{ij}}{dt} = R(t) \cdot e_{ij}(t) \quad (4.21)$$

gdje $R(t)$ određuje intenzitet i smjer promjene težine. Signal nagrade može doći i iz vanjskih izvora nakon pozitivnog događaja [162], ili se može izračunati kao pogreška u predviđanju nagrade (razlika između očekivanog i stvarnog ishoda) [163]. R-STDP omogućuje adaptivno učenje gdje pozitivna nagrada jača nedavno aktivne sinapse (potkrepljujući uspješne akcije), dok ih negativna nagrada slabi. Na taj način sustav uspješno rješava problem vremenske raspodjele zasluga, povezujući neuronsku aktivnost s ishodima koji se događaju nakon nekoliko milisekundi [105], [152], [154].

STDP sa maksimizacijom nagrade (engl. *Reward-Maximization*, R-max) i STDP vremenske razlike (engl. *Temporal-Difference STDP*, TD-STDP) dvije su inačice koje se također temelje na pravilu trećeg signala [152]. Dok R-STDP reagira na nagradu, R-max ju pokušava optimizirati [152], a TD-STDP uči na temelju razlike između očekivane i stvarno primljene nagrade [164]. Uz modele temeljene isključivo na nagradi, razvijeni su i hibridni pristupi koji kombiniraju vanjske signale i principe nadziranog učenja. ReSuMe (*Remote Supervised*

⁴ Kemijske tvari u živčanom sustavu koje moduliraju aktivnost neurona. Primjerice, dopamin djeluje kao signal koji može kodirati informaciju o nagradi, novosti ili pogrešci predviđanja te na taj način dugotrajnije pojačavati ili smanjivati aktivnost sinapse.

Method) spaja nadzirano učenje s potkrepljivanjem, prilagođavajući sinaptičke težine tako da se stvarni izlaz mreže uskladi s ciljanim odgovorom, uz vođenje signalom iz okoline [165]. Širenje podobnosti (engl. *Eligibility Propagation*, e-prop) predstavlja biološki uvjerljivu alternativu algoritmu povratnog širenja pogreške koji koristi tragove podobnosti i rješava problem raspodjele zasluga (engl. *credit assignment problem*)

Tako da svaka sinapsa privremeno pamti prethodnu aktivnost neurona, dok se globalni signal učenja pojavljuje kada se procijeni uspješnost ishoda. Sinapsa kombinira svoj lokalni trag aktivnosti s globalnim signalom učenja kako bi se odredio njezin doprinos kasnijem ishodu, čime se izbjegava potreba za memorijski zahtjevnim pohranjivanjem cjelokupne povijesti stanja mreže [166].

5 Implementacijski okviri i primjene SNN-a

Razvoj SNN-ova povezan je s potragom za računalnim arhitekturama koje nadilaze ograničenja tradicionalne Von Neumannove arhitekture. Dok klasični procesori (CPU) i grafičke kartice (GPU) troše energiju na stalni prijenos podataka između memorije i procesora, biološki sustavi taj problem rješavaju integracijom memorije i obrade unutar sinapse.

5.1 Neuromorfno inženjerstvo i hardverske realizacije

Neuromorfno inženjerstvo, termin koji je 1990-ih uveo Carver Mead, odnosi se na dizajn elektroničkih sustava čija arhitektura izravno oponaša neurobiološke strukture živčanog sustava [92]. Temeljna ideja je zamijeniti klasičnu digitalnu logiku temeljenu na sinkronom taktu sa sustavima koji rade na principu događaja kako bi se postiglo smanjenje energetske potrošnje i povećala učinkovitost obrade podataka. Ključna razlika u odnosu na tradicionalnu von Neumannovu arhitekturu je u organizaciji memorije i procesorske jedinice. U tradicionalnim računalnim sustavima one su odvojene, što dovodi do ograničene propusnosti poznatog kao von Neumannovo usko grlo, neuromorfni sustavi implementiraju memoriju unutar same mreže neurona i sinapsi. Time se omogućuje lokalna pohrana i obrada informacija, uz asinkronu komunikaciju temeljenu na prijenosu diskretnih impulsa (događaja), a ne kontinuiranih numeričkih vrijednosti. Zbog takvog načina obrade informacija, SNN-ovi su prikladni za implementaciju na neuromorfnim čipovima.

Tablica 5.1 Usporedba digitalnih i analognih/mješovitih neuromorfnih sustava.

Karakteristika	Digitalni sustavi	Analogni / mješoviti sustavi
Gustoća neurona	Umjerena (više tranzistora po neuronu)	Vrlo visoka
Prikaz stanja (potencijal membrane)	Diskretne binarne vrijednosti	Kontinuiran napon ili struja
Preciznost	Visoka i ponovljiva	Ograničena, osjetljivost na šum i temperaturu
Podешavanje parametara	Softverski podešivi	Djelomično određeni hardverom
Brzina	Realno vrijeme ili sporije	Moguće ubrzano izvođenje
Potrošnja energije	Veća, raste s veličinom sustava	Vrlo niska (najbliža biologiji)

Prema načinu implementacije neurona, neuromorfni hardver može se podijeliti u dvije skupine: digitalne i analogno-digitalne (mješovite) sustave [167]. Ključna razlika između njih očituje se u načinu na koji se matematički model neurona implementira u siliciju, odnosno u razlici između formalnog modela i njegove fizičke realizacije. Digitalni neuromorfni sustavi temelje se na standardnim elementima digitalne logike (logička vrata, registri, aritmetičko-logičke jedinice) gdje je stanje neurona, određeno potencijalom membrane, zapisano kao binarni broj u memoriji. Dinamika neurona opisuje se diferencijalnim jednadžbama kroz vremenske korake u kojima sustav akumulira ponderirane ulazne impulse i provjerava je li dosegnut prag okidanja. Analogni sustavi, s druge strane, ne opisuju dinamiku izravnim izračunom, već koriste fizikalna svojstva tranzistora za simuliranje diferencijalnih jednadžbi. Potencijal membrane prikazan je kao električni napon na kondenzatoru, dok struje kroz tranzistore predstavljaju sinaptičke ulaze. Kada napon dosegne zadani prag, generira se impuls, a cijeli proces proizlazi iz prirodnog ponašanja sklopa. Većina suvremenih analognih neuromorfnih čipova zapravo je mješovite arhitekture. Analogni dio koristi se za opis ponašanja neurona i sinapsi, dok je komunikacija između neurona realizirana digitalno, najčešće primjenom *Address Event Representation* (AER) protokola. U AER-u svaki generirani impuls kodira se adresom neurona i prenosi se

zajedničkom digitalnom sabirnicom [168]. Budući da je frekvencija generiranja impulsa relativno niska u odnosu na propusnost digitalnih komunikacijskih kanala, više neurona može dijeliti isti komunikacijski kanal.

IBM je s čipom TrueNorth [169] demonstrirao digitalnu implementaciju impulsnih neuronskih mreža. Sustav integrira milijun digitalnih neurona i 256 milijuna sinapsi organiziranih u 4096 jezgri. Arhitektura je optimizirana za događajno vođenu obranu uz iznimno nisku potrošnju energije (desetak mV pri tipičnom radu). Treniranje se radi izvan čipa, dok je sam čip primarno namijenjen izvođenju unaprijed treniranih mreža. Funkcionalno napredniji pristup realiziran je na platformi Loihi koju je razvio Intel. Za razliku od TrueNortha, Loihi [170] implementira programabilne mehanizme lokalne plastičnosti na čipu, uključujući varijante STDP-a i drugih adaptivnih pravila. Ovim se omogućuje neposredna prilagodba sinaptičkih težina bez potrebe za vanjskim procesiranjem, što je bitno za adaptivne sustave u stvarnom vremenu. Drugačiji digitalni koncept predstavlja SpiNNaker [171]. Umjesto specijaliziranih neuronskih sklopova, SpiNNaker koristi velik broj procesorskih jezgri koje su povezane komunikacijskom mrežom optimiziranom za prijenos impulsa. Ponašanje neurona radi se softverski, čime se postiže visoka fleksibilnost i mogućnost modeliranja biološki detaljnih mreža. U kategoriju digitalnih sustava može se smjestiti i Darwin Neural Processing Unit [172], koja koristi vremensko multipleksiranje kako bi manjim brojem fizičkih neurona oponašao veći broj logičkih neurona. Time se ostvaruje kompromis između hardverske složenosti i računalnog kapaciteta.

Za razliku od digitalnih implementacija, mješovite arhitekture koriste fizikalna svojstva tranzistora za implementaciju diferencijalnih jednadžbi neuronske dinamike. Platforma Neurogrid [173] implementira neuronske i sinaptičke sklopove analogno, dok se komunikacija odvija digitalno. Sustav omogućuje simulaciju velikih mreža u realnom vremenu uz vrlo nisku potrošnju energije, što ga čini posebno prikladnim za istraživanja u računalnoj neuroznanosti. Sličan pristup primijenjen je u sustavu Braindrop [174], koji integrira analogne dijelove neurona s programskim okvirom čime se povećava razina apstrakcije i ublažava problem heterogenosti analognih komponenti. Posebnu kategoriju predstavlja BrainScaleS [175] sustav koji implementira analogne neuronske sklopove na razini cijele silicijske pločice pa je rad neurona ubrzan u odnosu na realno vrijeme. Novija generacija BrainScaleS-2 dodaje složenije oblike ponašanja u dendritima neurona.

Usporedba navedenih sustava pokazuje da digitalne arhitekture naglašavaju skalabilnost i integraciju s postojećim sustavima, dok mješoviti sustavi nude veću biofizičku vjernost i energetska učinkovitost pri modeliranju kontinuirane dinamike.

Neuromorfni hardver temeljen na događajima ne predstavlja samo optimizaciju klasičnih arhitektura računanja, već novu paradigmu prema sustavima koji procesiraju informacije na način bliži biološkom mozgu, s lokalno pohranjenim sinaptičkim težinama i učinkovitom komunikacijom inspiriranom živčanim sustavom [176]. Kao ilustracija ovog pristupa, analognim čipovima s integriranim memorijskim izračunom uspješno su izvedeni zadaci računalnog vida uz točnost koja je suporediva s digitalnim sustavima, ali uz znatno nižu potrošnju energije i bez učestalog prijenosa podataka između memorije i procesora, čime se ublažava ograničenje poznato kao „usko grlo“ prijenosa podataka [177]. Ipak, ANN-ovi često ostvaruju bolje rezultate na standardnim GPU platformama jer su njihove operacije temeljene na gusto strukturiranim matričnim množenjima, za što su takve arhitekture optimirane [177]. S druge strane, i same strukture SNN-ova doživljavaju napredak. Nedavno je pokazano da je moguće trenirati duboke SNN modele do točnosti usporedive s ANN-ovima (92.81%), pritom postižući izuzetno nisku stopu impulsa po neuronu, što dodatno smanjuje energetska potrošnju [178].

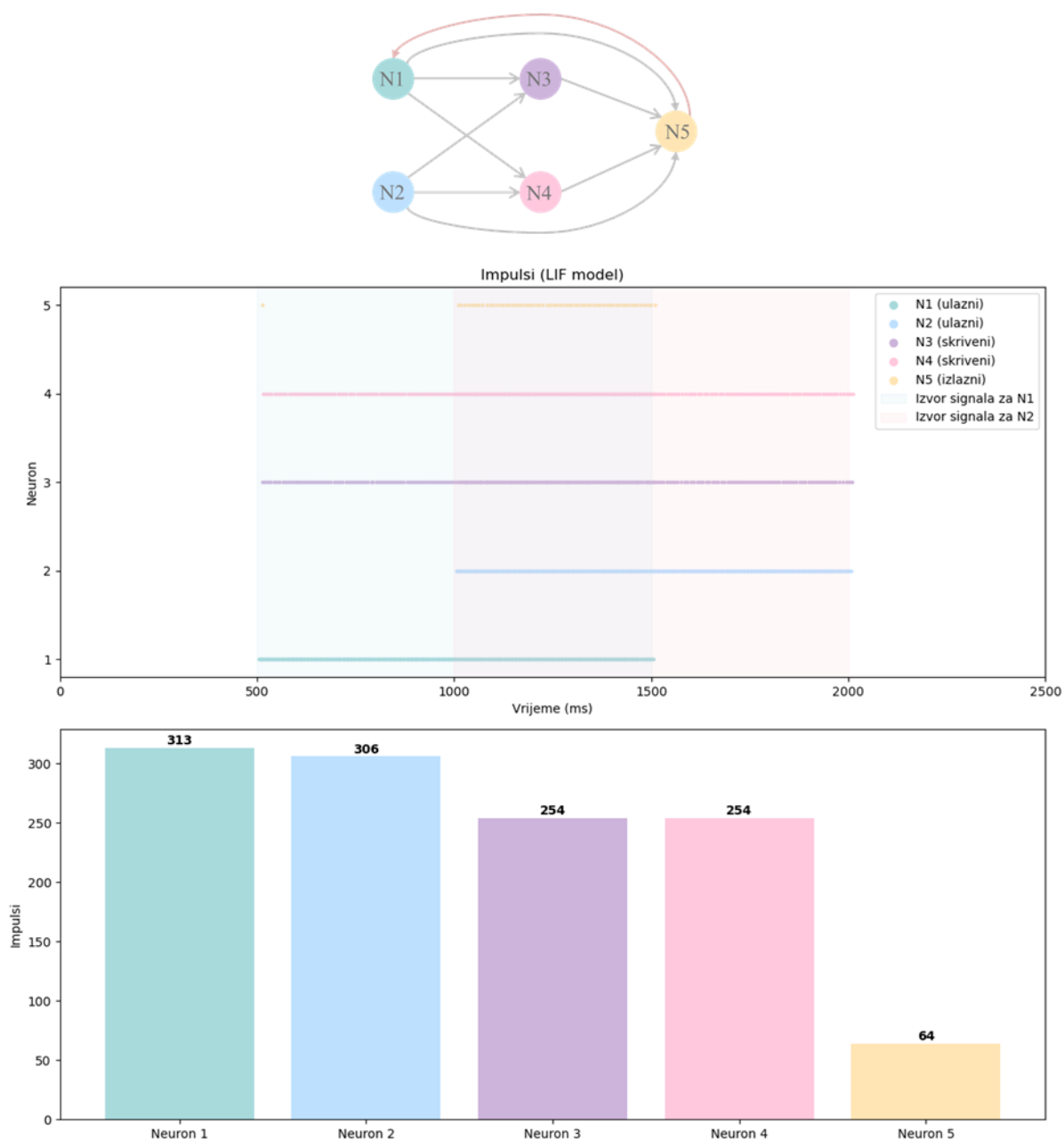
5.2 Programski okviri za simulaciju SNN-a

Razvoj softverskih alata za SNN-ove povijesno je bio primarno usmjeren na potrebe neuroznanosti i neurobiologije, gdje je naglasak na biološkoj vjerodostojnosti i simulaciji moždane aktivnosti. S rastućim interesom za primjenu neuromorfnog računarstva u području strojnog učenja, javlja se potreba za novim okvirima i alatima koji se mogu podijeliti u tri glavne kategorije: neuroznanstveno orijentirani simulatori, okviri za duboko učenje, i alati za konverziju i integraciju s neuromorfnim hardverom.

Neuroznanstveno orijentirani simulatori primarno su namijenjeni validaciji teorijskih modela usporedbom s eksperimentalnim podacima. Najvišu razinu biološke vjernosti pružaju NEURON [179] i GENESIS [180], koji su optimizirani za modele u kojima se neuron dijeli na više međusobno povezanih segmenata radi preciznog opisivanja (engl. *multi-compartment neuron model*). S druge strane, NEST je namijenjen za simulaciju velikih mreža koje se sastoje od pojednostavljenih modela neurona. Naglasak je na skalabilnost i distribuiranom izvođenju [181], zbog čega se često koristi u teorijskoj neuroznanosti, gdje nije potrebno modelirati unutarnju morfologiju pojedinog neurona. Fleksibilniji pristup modeliranju nudi Brian2 koji nudi opis neurona pomoću diferencijalnih jednažbi zapisanih u standardnom matematičkom obliku [182], a implementacija u Pythonu olakšava prilagodnu i proširenje modela. Na višoj razini apstrakcije, PyNN [183] nudi sučelje za modeliranje populacija neurona, sinapsi i njihove povezanosti. PyNN omogućuje prenosivost koda između sustava Brian2, NEST i NEURON i podržava izvođenje na neuromorfnim platformama kao što su SpiNNaker i BrainScaleS-2.

Svi ovi alati koriste simulaciju vođenu taktom (engl. *clock-driven simulation*), u kojoj se dinamika membrane opisuje numeričkim rješavanjem diferencijalnih jednažbi u diskretnim vremenskim koracima.

S razvojem području umjetne inteligencije javila se potreba za razvojem okvira koji omogućuju rad s SNN-ovima. Takvi okviri često koriste simulaciju temeljenu na događajima ili prilagođavaju SNN modele kako bi ih bilo moguće trenirati standardnim metodama. Alati kao što su SpikingJelly [184] i BrainCog [185] razvijeni su na temelju biblioteke PyTorch [186] te omogućuju treniranje dubokih SNN-ova uz akceleraciju na grafičkim procesorima. Norse [187] proširuje PyTorch implementacijama rekurentnih impulsnih mreža i algoritmima povratnog širenja pogreške kroz vrijeme. BindsNET [188] i SpykeTorch [189] usmjereni su na lokalna pravila učenja, uključujući STDP te metode potkrepljenog učenja. Dodatno, GeNN [190] generira programski kôd prilagođen konkretnom modelu mreže, što omogućuje učinkovitu simulaciju SNN-ova na grafičkim arhitekturama proizvođača NVIDIA.



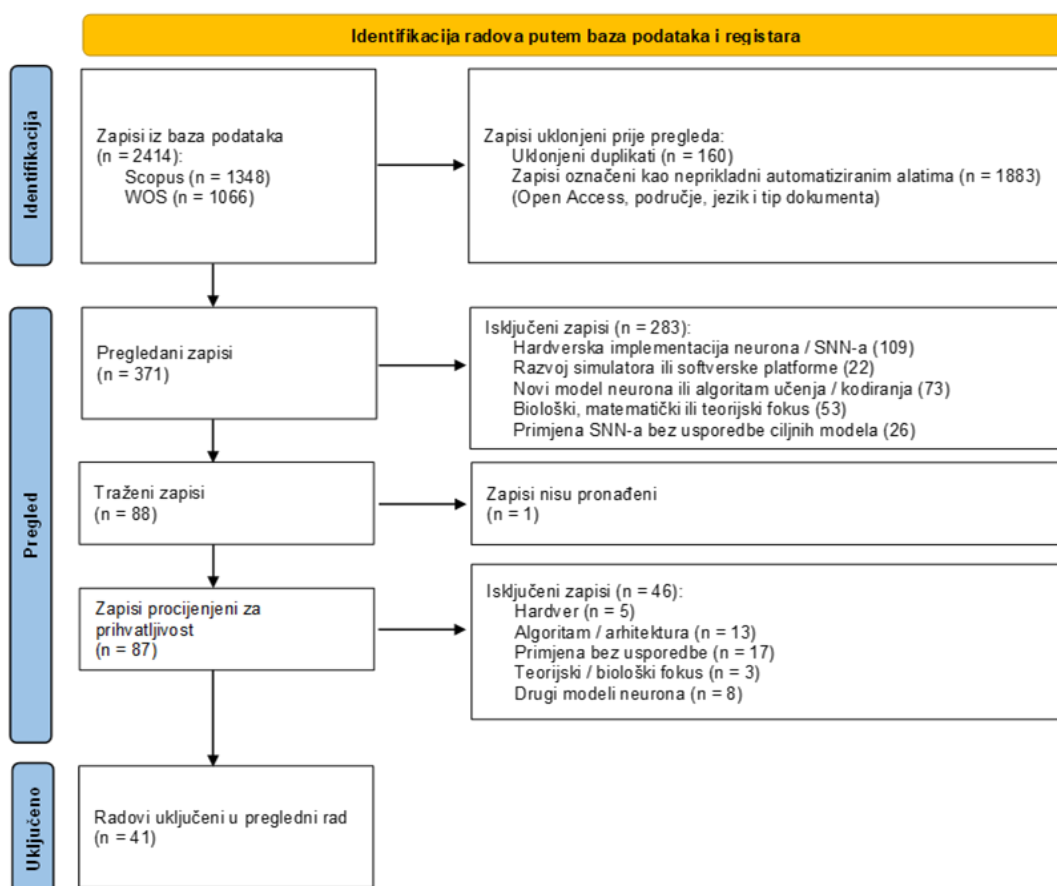
Slika 5.1 Simulacija aktivnosti XOR neuronske mreže temeljene na LIF modelu u NEST okruženju. Gornji panel prikazuje vremenski raspored akcijskih potencijala pet neurona tijekom 2500 ms. Osjenčana područja prikazuju aktivnost dvaju generatora impulsa (plavo: 500-1500 ms; rozo: 1000-2000 ms; 2 kHz), koji neuroni primaju (300 pA, 1 ms). Neuroni N1 i N2 su ulazni, N3 i N4 su skriveni i inhibiraju izlazni neuron N5 (-800 pA, 2 ms). N5 istodobno prima snažnu pobudu iz N1 i N2 (800 pA, 4 ms), ali je njegovo okidanje ograničeno inhibicijom. Prisutna je i slaba povratna veza N5→N1 (100 pA, 6 ms). Svi neuroni modelirani su kao iaf_psc_alpha ($\tau_m = 20$ ms, $C_m = 250$ pF, $V_{th} = -55$ mV).

Treću skupinu čine alati za konverziju ANN u SNN i njihovu integraciju s neuromorfnim sustavima. SNN Toolbox [191] transformira mreže treniranu u okruženju TensorFlow ili PyTorch u impulsne verzije pogodne za izvođenje na simulatorima ili neuromorfnim platformama. Nengo [192] omogućuje izgradnju kognitivnih sustava visoke razine apstrakcije, a putem modula NengoDL [193] moguće je povezivanje s TensorFlow-om i konverziju klasičnih mreža u impulsni oblik. Lava [194] je okvir otvorenoga koda namijenjen heterogenom računarstvu, osmišljen za povezivanje tradicionalnih procesorskih resursa s neuromorfnim procesorima kao što je Loihi 2.

6 Pregled područja

Najvažnija komponenta SNN sustava je model neurona. U literaturi se često koriste različiti modeli neurona koji se razlikuju prema razini biološke realističnosti i računalne složenosti, a među najčešće korištenim su Hodgkin-Huxley, Leaky Integrate-and-Fire (LIF) i Izhikevich neuron. Odabir modela neurona može značajno utjecati na arhitekturu mreže, računalne zahtjeve, mogućnosti treniranja te performanse SNN sustava u različitim primjenama. Zbog toga je proveden sustavni pregled literature s ciljem identifikacije relevantnih radova koji analiziraju odnos između modela neurona i karakteristika SNN sustava.

Sustavni pregled literature proveden je u skladu sa PRISMA (*Preferred Reporting Items for Systematic Reviews and Meta-Analyses*) [195] smjernicama, s ciljem identifikacije i analize znanstvenih radova koji istražuju utjecaj odabira modela neurona na arhitekturu, računalnu složenost i performanse impulsnih neuronskih mreža (SNN). Istraživanje je vođeno istraživačkim pitanjem: *Kako odabir modela neurona (Hodgkin-Huxley, Leaky Integrate-and-Fire i Izhikevich) utječe na arhitekturu, računalnu složenost i performanse SNN sustava?* Postupak identifikacije, filtriranja i odabira radova prikazan je u PRISMA dijagramu toka (Slika 6.1).



Slika 6.1 PRISMA dijagram toka odabira radova za pregled područja.

Pretraga literature provedena je 2. ožujka 2026. godine, pri čemu radovi objavljeni nakon tog datuma nisu uključeni u pregled. Pretraga je provedena u bazama Web of Science i Scopus sa

istim naprednim upitom pretraživanja: ("Spiking Neural Network*" OR SNN) AND ("Hodgkin-Huxley" OR HH) OR ("Leaky Integrate-and-fire" OR "leaky integrate and fire" OR LIF) OR (Izhikevich OR Izh)).

U Web of Science bazi pretraga je rezultirala s 1066 dokumenata, nakon čega je primijenjen filter *Open Access*, čime je broj radova smanjen na 376. Iz pregleda su isključeni radovi označeni kao *Early Access* (3) i *Correction* (2). Dodatno su primijenjeni tematski filtri koji su obuhvatili područja računalnih znanosti, umjetne inteligencije, informacijskih sustava, teorijskih metoda, multidisciplinarnih znanosti, računalne arhitekture, robotike i srodnih inženjerskih disciplina, čime je konačan broj radova iz ove baze smanjen na 172. U Scopus bazi pretraga je vratila 1348 dokumenata, od kojih je filtriranjem na *Open Access* preostalo 426 radova. Iz pregleda su uklonjeni zapisi tipa *Letter* (3), *Erratum* (2) i *Note* (1), a dodatno su zadržani samo radovi objavljeni na engleskom jeziku, čime je broj smanjen na 410 radova. Primjenom filtra prema područjima istraživanja (računarstvo, inženjerstvo, neuroznanost i multidisciplinarnost) konačan broj radova iz baze Scopus iznosio je 359. Inicijalnom pretragom identificiran je 531 rad, a nakon uklanjanja 160 duplikata, za daljnju analizu ostao je 371 rad. Na tim radovima proveden je postupak probira prema naslovu i sažetku, pri čemu je 283 rada isključeno zbog neusklađenosti s ciljem istraživanja. Razlozi isključivanja uključivali su radove usmjerene na hardversku implementaciju neurona ili SNN sustava (109), razvoj simulatora ili softverskih platformi (22), radove čiji je primarni doprinos bio novi model neurona ili algoritam učenja / kodiranja (73), radove s biološkim, matematičkim ili teorijskim fokusom (53) te radove koji su opisivali primjene SNN sustava bez usporedbe ciljanih modela neurona (26). U završnoj fazi provedena je detaljna analiza 88 radova. Nakon čitanja, dodatnih 47 radova je isključeno iz pregleda jer nisu zadovoljavali kriterije uključivanja. Razlozi isključenja uključivali su radove fokusirane na hardversku realizaciju neurona ili neuromorfni sustava bez usporedbe modela neurona (5), radove čiji je primarni doprinos bio novi algoritam učenja ili arhitektura mreže (13), radove koji su koristili SNN u specifičnim primjenama bez analize utjecaja modela neurona (17), radove s biološkim ili teorijskim fokusom (3), radove koji nisu koristili tražene modele neurona ili su razvijali nove (8) te jedan rad koji nije bio dostupan u punom tekstu. Nakon završetka postupka selekcije, 41 znanstveni rad je uključena u završnu analizu.

Kroz radove je vidljiv širok spektar primjena, uključujući klasifikaciju slika, medicinsku i industrijsku dijagnostiku, navigaciju autonomnih robota te razvoj sustava namijenjenih izvršavanju na *edge* i IoT uređajima. Korišteni modeli neurona, eksperimentalni protokoli i korišteni skupovi podataka znatno se razlikuju između radova, što otežava izravnu usporedbu, ali omogućuje sintezu o utjecaju modela neurona u različitim kontekstima.

6.1 Zastupljenost modela neurona

Pregledom analiziranih radova utvrđena je dominacija LIF modela koji je zastupljen u 37 od ukupno 41 rada. Izhikevich model je korišten u pet radova, H-H model nije zabilježen u nijednom analiziranom radu, a jedan rad uspoređuje LIF i Izhikevich model. U pojedinim radovima korištene su i varijante navedenih modela (QIF, EIF, IF, klasični umjetni neuron...), pri čemu su one najčešće bile uključene u svrhu usporedne analize s jednim od triju modela koji su bili u primarnom fokusu istraživanja. Hodgkin-Huxley model zastupljen je u analiziranom skupu radova isključivo u obliku referenci ili kontekstualnih napomena unutar preglednih radova ali nijedan rad iz analiziranog skupa ne provodi eksperimentalnu evaluaciju H-H modela u okviru SNN sustava. Sukladno tome, zaključci koji se tiču H-H modela temelje se na teorijskim svojstvima poznatim iz literature, a ne na empirijskim podacima iz analiziranog skupa.

Detaljna distribucija modela, zajedno s njihovim tipičnim primjenama i karakterističnim obilježjima, prikazana je u Tablica 6.1.

Tablica 6.1 Distribucija modela neurona u analiziranoj literaturi s pregledom primjena i obilježja.

Model neurona	Broj radova	Tipične primjene	Prednosti	Ograničenja
LIF	37	Klasifikacija slika, medicinska dijagnostika, <i>edge</i> uređaji	Jednostavnost, hardverska kompatibilnost, brze simulacije	Ograničena ekspresivnost za složene vremenske uzorke; osjetljivost na hiperparametre
Izhikevich	5	Robotika, učenje potkrepljivanjem, prepoznavanje govora, industrijska dijagnostika	Bogata neuronska dinamika, impulsna adaptacija, reprodukcija više tipova neuronskog ponašanja	2-2.5 puta veći računalni trošak od LIF-a; složenija implementacija
H-H	0		-	

Sveprisutnost LIF modela u analiziranim radovima može se objasniti s nekoliko razloga. Prvi je njegova matematička jednostavnost, koja omogućuje lakšu obradu i simulaciju neurona. Drugi razlog je njegova pogodnost za primjenu na neuromorfnom hardveru, pri čemu niža računalna složenost doprinosi većoj energetske učinkovitosti sustava [196], [197]. Treći čimbenik je dobra kompatibilnost s često korištenim razvojnim okvirima za strojno učenje, pri čemu je PyTorch zastupljen u gotovo polovici analiziranih radova. U području konverzije umjetnih neuronskih mreža u impulsne mreže (ANN→SNN), LIF model se često koristi upravo zbog svoje prilagođenosti hardverskoj implementaciji. Iako postoje biološki precizniji modeli, njihova je primjena rjeđa zbog većih zahtjeva, dok LIF model i dalje pruža povoljan odnos između računalne složenosti i točnosti za većinu standardnih zadataka strojnog učenja [198].

6.2 Utjecaj modela neurona na performanse i složenost SNN sustava

Složenost modela neurona i utjecaj na arhitekturu

U analiziranoj literaturi, složenost modela neurona pokazuje se kao ključan čimbenik koji utječe na SNN arhitekturu. Posebno se ističe LIF model, čija relativno jednostavna dinamika omogućuje skaliranje mreža na velik broj slojeva.

Pokazano je da LIF mreže mogu biti trenirane u arhitekturama dubine do 68 slojeva bez degradacije performansi, što je znatno teže postići s kompleksnijim neuronskim modelima [199]. U eksperimentalnim evaluacijama na MNIST skupu LIF-temeljene mreže ostvaruju vrlo visoku klasifikacijsku točnost od 99.59%, čime se postižu vrhunski rezultati (engl. *State of the Art*, SOTA) [200]. Poboľšanjem modela kroz koeficijent curenja membrane, točnost se podiže na 99.67% na istom skupu podataka [201]. Ovi rezultati ukazuju na to da LIF model, unatoč relativno jednostavnom opisu neuronske dinamike, može ostvariti SOTA rezultate. Jednostavnost LIF modela dodatno olakšava integraciju u različite arhitekturne pristupe i metode učenja, uključujući dinamičke arhitekture koje automatski prilagođavaju broj neurona. Razvijeni su modeli s troslojnom arhitekturom koja dinamički raste i reducira broj neurona unutar jedne epohe, čime se postiže jednaka ili viša točnost uz znatno manji broj parametara [202]. Organizacija neurona u funkcionalne ansamble inspirirane organizacijom u kortikalnom dijelu mozga koristi klasno-ovisne neuronske aktivacije za selektivno jačanje ili slabljenje

sinapsi primjenom STDP i anti-STDP pravila učenja, uz smanjenje parametara za 70-99% [203]. Integracija LIF neurona u standardne arhitekture strojnog učenja olakšana je upotrebom surogatnih gradijenata implementiranih u proširenju za *Keras*. Ovim se omogućuje razvoj hibridnih modela i ANN u SNN arhitekture unutar uobičajenih razvojnih okvira dubokog učenja [204]. Dodatni aspekt učinkovitosti LIF mreža odnosi se na energetske potrošnje, pri čemu događajno vođeni pristup u kombinaciji s lateralnom inhibicijom i WTA mehanizmom postiže približno 85% točnosti na skupu MNIST uz smanjenu potrošnju energije u odnosu na modele vođene taktom [205].

Za razliku od LIF modela, Izhikevičev model koristi dvije diferencijalne jednadžbe, što omogućuje modeliranje raznih tipova neuronskog okidanja (*bursting, chattering, fast spiking* i dr.). Takva fleksibilnost povećava biološku točnost modela i omogućuje učinkovitije prikupljanje informacija u zadacima koji zahtijevaju obradu vremenskih uzoraka ali dolazi uz viši računalni trošak, koji je procijenjeno na 2 do 2.5 puta veći u odnosu na LIF model [206]. Unatoč tome, Izhikevičev model pokazuje značajan potencijal u zadacima koji zahtijevaju adaptivno i vremenski ovisno učenje. U navigaciji autonomnog robota, SNN koji kombinira Izhikevičev neurone s STDP učenjem postiže 94% uspješnosti dosezanja cilja u usporedbi s početnih 33% [207]. Sposobnost obrade vremenski strukturiranih signala demonstrirana je i u području obrade govora, gdje se Izhikevičev neuroni koriste za ekstrakciju značajki impulsa iz zvučnih signala te omogućuju uspješnu klasifikaciju izgovorenih znamenki primjenom STDP pravila u Hebbovom i anti-Hebbovom obliku [208]. Implementacija impulsne konvolucijske mreže temeljene na Izhikevičev modelu na FPGA platformi korištena je za detekciju tumora na MRI snimkama, pri čemu se klasifikacija temelji na frekvenciji okidanja neurona u izlaznom sloju [209]. U kontekstu dijagnostike mrežnih anomalija na *edge* uređajima razvijena je mreža s dinamičkim STDP-om, i postiže F1-mjeru od 0.98 na SMD skupu podataka, 0.96 na ASD skupu i pokazuje bolje rezultate nad LIF modelima [210].

Za razliku od prethodna dva modela, Hodgkin-Huxley model predstavlja biofizikalno najrealističniji opis neuronske dinamike među razmatranim pristupima ali u analiziranim radovima ovaj model nije evaluiran u kontekstu performansi SNN-ova. Navodi se kroz pregledni rad ili u kontekstu evaluacije neuronskih modela, kao referentni model, ali bez empirijske procjene njegove učinkovitosti u zadacima strojnog učenja [197], [211]. Teorijski gledano, četiri nelinearne jednadžbe H-H modela čine njegovu primjenu u velikim mrežama računalno nepraktičnim, što vjerojatno objašnjava izostanak eksperimentalne evaluacije. Sličan zaključak navodi se i u kontekstu konverzije ANN mreža u SNN-ove, gdje se ističe da su biološki detaljniji modeli često preskupi za implementaciju u realnim sustavima [198]. Stoga se u okviru ove analize zaključci o primjenjivosti H-H modela mogu formulirati prvenstveno na teorijskoj razini: zbog potrebe za numeričkom integracijom više nelinearnih jednadžbi njegova je računalna zahtjevnost veća od LIF i Izhikevičev modela, što ograničava njegovu praktičnu primjenu u velikim SNN arhitekturama i na *edge* uređajima.

Računalni trošak i energetska učinkovitost

Energetska učinkovitost jedna je od glavnih motivacija za razvoj SNN-ova, gdje model neurona izravno određuje računalnu cijenu kroz svaki impuls. Složenost neuronske dinamike određuje broj aritmetičkih operacija potrebnih za simulaciju potencijala membrane i sinaptičkih procesa, što se izravno odražava na potrošnju energije sustava. U analiziranoj literaturi ova je dimenzija evaluirana isključivo za LIF model neurona, zbog čega izravna empirijska usporedba s Izhikevičev i Hodgkin-Huxley modelima u jednakim implementacijskim uvjetima nije dostupna.

Kvantitativni dokazi o energetske učinkovitosti SNN arhitektura najčešće se odnose na implementacije temeljene na LIF modelu neurona [196], [197], [212]. Analogna CMOS

implementacija LIF neurona ostvaruje potrošnju od približno 1.06 mW, što je oko dvadeset puta manje u odnosu na digitalnu implementaciju iste arhitekture [196]. Dodatne optimizacije na razini modela neurona i hardverske implementacije omogućuju smanjenje potrošnje resursa. Primjenom pojednostavljenog LIF modela postiže se približno 50% manja potrošnja hardverskih resursa i oko 39% brže odzivno vrijeme u odnosu na uspoređene modele neurona [197]. Općenito, dostupne analize pokazuju da SNN modeli ostvaruju približno 90-97% nižu energetska potrošnju u odnosu na klasične ANN arhitekture. Prednost SNN pristupa posebno je izražena u usporedbi s konvencionalnim ANN i CNN modelima u zadacima medicinske dijagnostike, gdje je zabilježeno smanjenje potrošnje energije od približno 986 puta (0.192 J naspram 186 J po predikciji), uz smanjenje klasifikacijske točnosti za 0.4% [212], a pristup temeljen na STDP pravilu na klasičnim skupovima podataka troši približno 5 mJ uz maksimalnu klasifikacijsku točnost od oko 95% [12].

Daljnje optimizacije LIF modela posebno su izražene u radovima usmjerenima na razvoj *laganih* arhitektura za *edge* računalne sustave. Jedan pristup kombinira STDP pravilo i mehanizam lateralne inhibicije. Računalna složenost procesa učenja smanjila se s $O(n^2)$ na $O(1)$ i postigla se točnost od 96.6% na MNIST skupu [213]. Arhitektura poput *tiny dLIF*, koristi prošireni LIF model s dendritičkim granama, pri čemu svaki neuron sadrži četiri dendritičke grane za obradu ulaznih signala. Takva arhitektura sadrži manje od 300 000 parametara i zahtijeva manje od 1.5 MB memorije i postiže izvrsne rezultate u detekciji epileptičkih napadaja na EEG signalima [214]. Dodatno, kontrolirano uvođenje šuma u neuronu može poboljšati točnost u rješavanju linearno neseparabilnih problema, poput XOR zadatka, gdje SNN nadmašuje klasične ANN modele ograničene ReLU aktivacijskim funkcijama za više od 10% [215].

Nasuprot tome, podaci o računalnom trošku Izhikevičevog modela su oskudni. Dostupni izvori navode da ovaj model zahtijeva 2 do 2.5 puta veći trošak po iteraciji u usporedbi s LIF-om [206]. Iako postoje implementacije Izhikevičevog modela na FPGA platformama, često izostaju numeričke metrike o stvarnoj energetska potrošnji na hardveru [209].

Točnost i primjenjivost u različitim domenama

Analiza literature otkriva povezanost između tipa podataka i relativne prednosti specifičnih modela neurona, ali optimalna izvedba SNN-a je rezultat međudjelovanja modela neurona, pravila učenja i karakteristika signala, zbog čega ne postoji univerzalno rješenje za sve zadatke [216].

Na skupovima statičkih slika (MNIST, Fashion MNIST, CIFAR-10), LIF model postiže rezultate usporedive s umjetnim neuronskim mrežama uz znatno manju potrošnju energije. Na MNIST skupu podataka, LIF neuroni postižu točnost u rasponu od 90% do gotovo savršene klasifikacije primjenom različitih metoda treniranja, uključujući ANN u SNN pretvorbu, surogatne gradijente i izravnu propagaciju pogreške unatrag. Slična učinkovitost zabilježena je i na kompleksnijim skupovima poput CIFAR-10 i CIFAR-100, gdje biološki inspirirani mehanizmi pažnje omogućuju postizanje visokih performansi u svega četiri vremenska koraka [198], [199], [200], [201], [205], [217], [218], [219], [220].

Međutim, na neuromorfni skupovima podataka⁵ s izraženom vremenskom dinamikom, prednost uzimaju kompleksniji modeli. Izhikevičev model i srodne varijante (poput QIF i EIF) nadmašuju LIF na DVS Gestures skupu podataka, što se pripisuje nelinearnosti potencijala membrane koja bolje kodira temporalne uzorke visoke varijabilnosti [109]. Dokaz nadmoći

⁵ Neuromorfni skupovi podataka oponašaju način na koji biološki neuronski sustavi obrađuju informacije, najčešće u obliku vremenski kontinuiranih, događajno vođenih signala umjesto klasičnih statičkih uzoraka.

Izhikevičevog modela vidljiv je i na klasičnim klasifikacijskim skupovima (Iris, Wine, Breast Cancer) i kod učenja potkrepljivanjem gdje LIF model često ne uspijeva pronaći rješenje [206]. Ipak hibridni pristupi koji mapiraju trenirane ANN modele na LIF SNN ili rekurentne LIF mreže trenirane sa STDP-om, uspijevaju zadržati visoku točnost uz minimalne gubitke na DVS skupovima podataka [198], [109], [220], [221].

Izbor strategije kodiranja signala dodatno utječe na performanse LIF mreža: korištenjem kontinuiranih vrijednosti dobiva se najviša točnost ali i najviši energetska trošak, kodiranje latencijom nudi optimalnu energetska učinkovitost, dok kodiranje frekvencijom pruža najbolju robusnost i otpornost u napadima na ulazne podatke [223]. Posebna prednost SNN-a u odnosu na konvolucijske mreže očituje se pri radu s ograničenim skupovima podataka. Na svega 100 slika za treniranje SNN postiže točnost iznad 90%, dok klasični modeli zahtijevaju znatno veće arhitekture i resurse [224]. Također, LIF modeli s inhibicijskim neuronima pokazuju značajan napredak u odnosu na standardne STDP modele [219] što sugeriraju da je LIF model, uz adekvatnu metodu učenja, dostatan za klasifikacijske zadatke u kojima dominiraju prostorne značajke.

U medicinskim primjenama, koje čine četvrtinu analizirane literature, LIF model je najzastupljeniji. Visoka točnost postignuta je u klasifikaciji EKG signala, pri čemu SNN modeli često nadmašuju ANN modele, osobito kada se koristi mehanizam pažnje [225], [226]. Usporedne analize pokazuju da modeli poput AdEx-a mogu pružiti višu opću točnost, ali LIF model zadržava višu osjetljivost, što minimizira propust pozitivnih slučajeva u klasifikaciji EKG signala [227]. Primjena LIF arhitektura u 3D detekciji plućnih nodula na CT snimkama postiže bolje rezultate usporedive s CNN-om, ali uz 60% manje računalnih operacija [228]. Dobri rezultati zabilježeni su i u klasifikaciji Alzheimerove bolesti, gdje hibridni CNN / SNN modeli postižu točnost od 99.58% u odnosu na CNN model koji ostvaruje 75.67% [229]. Slična učinkovitost primijećena je u detekciji epileptičkih napada i depresije iz EEG signala, gdje SNN modeli postižu visoku točnost i bolje rezultate nad klasičnim metodama strojnog učenja [230], [231].

Primjene SNN-ova proširuju se i na industrijske sustave. Hibridna IF-LIF arhitektura, pri čemu konvolucijski dio koristi IF neurone, a klasifikacijski dio LIF neurone, postiže gotovo apsolutnu točnost u dijagnostici kvarova industrijskih ležajeva, nadmašujući uspoređenu CNN mrežu [232]. U detekciji objekata, hibridne SNN mreže pokazuju veću robusnost u uvjetima jakog šuma u usporedbi s konvencionalnim modelima poput RetinaNet-a [233], a u kontekstu kibernetičke sigurnosti, SNN modeli trenirani izravnom propagacijom pogreške unatrag pokazuju bolju otpornost na *blackbox* napade u odnosu na ANN modele, dok robusnost na *whitebox* napade izravno ovisi o primijenjenoj metodi kodiranja [220], [223].

6.3 Analiza sličnosti i razlika

Analizirani radovi pokazuju relativne usklađenosti u određenim pitanjima i izražene razlika u drugima. Tablica 6.2 sažima identificirane uzorke konvergencije i divergencije.

Tablica 6.2 Identificirani uzorci konvergencije i divergencije u analiziranoj literaturi.

	Tvrdnja / nalaz	Radovi
Konvergencija	LIF postiže SOTA točnost na MNIST podacima	[198], [199], [200], [204], [211], [212], [217], [218], [222], [223]
	SNN troše 90-986 puta manje energije od ekvivalentnih ANN modela uz malen pad točnosti	[12], [196], [212]
	STDP postiže nižu točnost od surogatnog gradijenta i ANN→SNN metoda	[12], [202], [219], [221], [233], [234]

	Izhikevich nadmašuje LIF na kompleksnim, temporalno bogatim zadacima	[109], [206], [210]
Divergencija	Univerzalna LIF primjena	[211] / [206]
	ANN→SNN konverzija bez gubitka točnosti	[198] / [12], [234]

Najveći konsenzus u analiziranoj literaturi odnosi se na energetska učinkovitost SNN-ova temeljenih na LIF modelu. U usporedbi s ekvivalentnim modelima umjetnih neuronskih mreža, SNN-ovi ostvaruju smanjenje potrošnje energije od jednog do tri reda veličine [12], [195], [199], [210], [211], [225], [232]. Navedena prednost izravna je posljedica rijetke aktivacije karakteristične za impulsne mreže, a nalaz je neovisan o arhitekturi, skupu podataka ili metodi treniranja. Drugo područje suglasja tiče se slabije učinkovitosti STDP metode treniranja u usporedbi sa nadziranim metodama učenja. Mreže temeljene na STDP pravilu postižu nižu točnost klasifikacije u odnosu na metode surogatnog gradijenta ili konverziju iz ANN u SNN [12], [226], [233], [234]. Primjerice, dok nadzirani pristupi na MNIST skupu ostvaruju točnost od 97% do 99.67% [199], [200], [201], [218], STDP-bazirane mreže postižu znatno skromnije rezultate u rasponu od 88.5% do najviše 96.6% [202], [219], [221], [224]. Ovi podaci sugeriraju da je izbor metode treniranja važniji od samog izbora između srodnih modela neurona. Treći konsenzus naglašava prednost kompleksnijih neuronskih modela u uvjetima bogatih vremenskih uzoraka. Istraživanja neovisno potvrđuju da nelinearna dinamika membrane Izhikevichevog (i srodnih QIF i EIF) modela pruža jasnu prednost nad LIF modelom u zadacima koji zahtijevaju složenu vremensku obradu ili rješavanje problema u potkrepljenom učenju, gdje LIF modeli često potpuno zakazuju [109], [206], [210].

Značajna divergencija u literaturi odnosi se na pitanje univerzalne primjenjivosti LIF modela. Dok određeni pregledni radovi tretiraju LIF kao zadovoljavajući standard za postizanje vrhunskih rezultata na statičkim skupovima poput MNIST-a i CIFAR-10 [211], drugi radovi eksplicitno demonstriraju njegove limite u dinamičkim okruženjima [206]. Ova razlika ne predstavlja proturječje, već pokazuje da zaključci ovise o specifičnoj primjeni. Druga točka razilaženja tiče se postupka konverzije ANN u SNN. Određeni radovi demonstriraju konverziju bez gubitka točnosti uz minimalan broj vremenskih koraka [198], dok druge izvještavaju o značajnom padu performansi u korist izravno treniranih SNN mreža [12], [234]. Razlike vjerojatno proizlaze iz tehnika konverzije. U medicinskom kontekstu, divergencija se očituje u izboru modela neurona. Dok većina radova koristi LIF model u klasifikaciji EKG-a [225], [226], [230], [231], usporedbe s kompleksnijim modelom pokazuju da izbor ovisi o evaluacijskoj metrici. U jednoj takvoj usporedbi složeniji adaptivni model postiže višu ukupnu točnost, dok LIF model ostvaruje višu vrijednost osjetljivosti, što ga čini pogodnijim za scenarije u kojima je kritično minimizirati broj propuštenih pozitivnih slučajeva [227].

Ukupno gledano, analiza divergencija pokazuje da većina uočenih razlika između studija nije posljedica empirijskih dokaza, već prije svega razlika u eksperimentalnim uvjetima. Različiti skupovi podataka, arhitekture mreža, metode treniranja i evaluacijske metrike značajno utječu na dobivene rezultate. Stoga se divergencije u literaturi mogu tumačiti prvenstveno kao posljedica metodološke heterogenosti i nedostatka standardiziranih metoda evaluacije, a ne kao dokaz međusobno proturječnih empirijskih nalaza.

6.4 Analiza ograničenja i nedostataka u postojećoj literaturi

Analiza radova obuhvaćenih ovim pregledom otkriva nekoliko strukturnih i metodoloških nedostataka koji ograničavaju donošenje zaključaka o optimalnosti pojedinih modela neurona.

Nedostatak komparativnih studija i metodološka homogenost

Temeljni problem predstavlja izrazit deficit istraživanja koja provode izravnu usporedbu dvaju ili više modela neurona pod identičnim eksperimentalnim uvjetima. Među analiziranim radovima, svega četiri studije provode takvu evaluaciju, pri čemu nijedna ne uključuje sva tri ključna modela (LIF, Izhikevich, H-H), a H-H model nije eksperimentalno evaluiran ni u jednom analiziranom radu [109], [206], [216], [227]. Radovi koji izvještavaju o najvišim razinama točnosti isključivo se oslanjaju na LIF model bez razmatranja alternativa [198], [199], [200], [201], [212], [217], [221], [234] što onemogućuje zaključivanje o tome je li visoka točnost rezultat superiornosti LIF modela ili pak specifičnih arhitektura i metoda treniranja. Identificirana je jasna potreba za sustavnim studijama koje bi, uz fiksirane parametre arhitekture i učenja, varirale isključivo model neurona kako bi se izolirao stvarni doprinos performansama.

Status Hodgkin-Huxley i Izhikevich modela

Posebno izražena praznina u literaturi odnosi se na izostanak eksperimentalne evaluacije H-H modela. Navedena praznina u literaturi ostavlja neistraženo područje potencijalne primjene H-H modela u domenama gdje su biološka vjerodostojnost i objašnjivost od primarne važnosti, poput modeliranja neurodegenerativnih bolesti.

Slično tome, Izhikevichev model se sustavno koristi u kombinaciji sa algoritmima učenja poput STDP-a ili evolucijskih algoritama [206], [207], [208], [210], dok LIF model dominira u istraživanjima koja primjenjuju moderne surogatne gradijente [199], [200], [201], [204], [217], [234]. Ipak, ključni iznimka i važan empirijski dokaz nalazi se u radu koji provodi izravnu zamjenu LIF modela Izhikevichevim modelom unutar iste arhitekture [210]. Unatoč ovom nalazu, on ostaje usamljen primjer u literaturi. Većina studija i dalje ne istražuje sustavno interakciju između modela neurona i modernih optimizacijskih tehnika. Nedostatak usporedbi pod jednakim uvjetima učenja (npr. primjena surogatnih gradijenata na Izhikevichev model u širem spektru zadataka) i dalje onemogućuje potpuno razumijevanje korelacije između biofizičke detaljnosti i performansi mreže u različitim domenama primjene.

Ograničenja evaluacijskih skupova podataka

Većina istraživanja i dalje daje prednost MNIST skupu podataka. Iako su SNN modeli prirodno pogodni za neuromorfne skupove podataka, takvi se skupovi rijetko pojavljuju u literaturi, što upućuje na sklonost korištenju jednostavnih referentnih skupova podataka za prikaz novih metoda. Međutim, takvi skupovi nisu dovoljni za pouzdanu procjenu sposobnosti modela da obrađuju informacije koje se mijenjaju kroz vrijeme, što je važno za razlikovanje jednostavnijih i složenijih pristupa [109], zbog rjeđe zastupljenosti teško je donošenje zaključaka o performansama modela u realističnijim uvjetima.

U medicinskim primjenama, evaluacija se u pravilu ograničava na jedan specifičan skup podataka, što značajno ograničava mogućnost generalizacije dobivenih rezultata [209], [210], [214], [225], [227], [228], [229], [231]. Nedostatak validacije na više skupova podataka dodatno smanjuje pouzdanost zaključaka o praktičnoj primjeni predloženih modela.

Manjak empirijskih podataka o energetskej potrošnji

Iako se energetska učinkovitost navodi kao primarna motivacija, kvantitativna mjerenja stvarne potrošnje na hardveru dostupna su isključivo za LIF model [196], [197], [212], [226]. Za Izhikevichev model, unatoč hardverskim implementacijama, nedostaju numeričke metrike potrošnje energije, što predstavlja veliku prazninu [209]. Bez izravnih empirijskih mjerenja specijaliziranim hardverskim platformama za SNN simulaciju i implementaciju, tvrdnje o energetskej nadmoći LIF-a nad kompleksnijim modelima ostaju na razini teorijskih pretpostavki temeljenih na jednostavnosti jednadžbi, a ne na dokazima.

7 Zaključak

Ovaj kvalifikacijski doktorski ispit obuhvatio je razvoj modela neurona, od *klasičnih* umjetnih do biološki inspiriranih modela te analizirao njihove implikacije na arhitekturu i performanse impulsnih neuronskih mreža. Kroz pregled razvoja ovog područja može se zaključiti da su *klasični* modeli umjetnih neurona, unatoč velikom uspjehu u praksi, nedostatni jer ne uzimaju u obzir vrijeme kao važan dio obrade informacija te troše znatne količine energije. Njihova implementacija na Von Neumannovoj arhitekturi dodatno naglašava problem razdvojenosti memorije i procesiranja, što postaje kritično u kontekstu suvremenih zahtjeva za skalabilnošću i energetsom učinkovitošću. Suprotno tome, biološki modeli neurona uvode vremensku dimenziju kao ključni nositelj informacije, omogućujući znatno bogatije i realističnije modeliranje neuronske dinamike. Analiza biofizičkih modela, uključujući Hodgkin-Huxley, Izhikevich i Leaky Integrate-and-Fire modele, pokazuje postojanje kompromisa između biološke vjernosti i računalne složenosti. Dok Hodgkin-Huxley model najvjernije opisuje stvarno ponašanje neurona, njegova složenost otežava primjenu u većim sustavima. Jednostavniji modeli poput LIF-a omogućuju lakšu implementaciju i rad na većim mrežama, ali uz manju preciznost. Izhikevich model predstavlja dobar balans jer može opisati različite obrasce ponašanja neurona uz relativno malu računalnu složenost.

U kontekstu impulsnih neuronskih mreža, izbor modela neurona izravno utječe na ključne aspekte sustava: arhitekturu mreže, način prikaza informacija, metode učenja te ukupnu potrošnju resursa. SNN-ovi prirodno koriste rijetke i događajno uvjetovane signale, što može značajno smanjiti potrošnju energije, posebno u kombinaciji s neuromorfim hardverom. Međutim, analiza literature pokazuje da se te prednosti često ne iskorištavaju u potpunosti, ponajviše zbog korištenja neprikladnih skupova podataka i mjernih kriterija koji ne uzimaju u obzir vremensku prirodu ovih modela. Problem učenja u SNN-ovima i dalje je jedno od glavnih otvorenih pitanja. Iako postoje različiti pristupi, poput lokalnih pravila učenja, prilagodbi gradijentnih metoda i učenja potkrepljivanjem, još uvijek ne postoji jedinstven i općeprihvaćen način treniranja koji bi bio jednako učinkovit kao kod klasičnih dubokih mreža. Dodatni problem predstavlja to što su pristupi u literaturi međusobno vrlo različiti, što otežava usporedbu rezultata između radova.

Pregledom područja istraživanja uočeno je da mali broj radova provodi izravne i dosljedne usporedbe različitih modela neurona u istim uvjetima, a pritom se najdetaljniji biofizički modeli rijetko uključuju u takve analize. Zbog toga je teško donijeti jasne zaključke o njihovim prednostima i nedostacima. Buduća istraživanja trebala bi uključivati sustavne usporedbe svih triju modela neurona (Hodgkin-Huxley, Izhikevich i LIF) uz korištenje šireg skupa mjernih kriterija, poput potrošnje energije, brzine obrade i otpornosti na šum, a ne samo točnosti.

Impulsne neuronske mreže predstavljaju obećavajući smjer razvoja umjetne inteligencije, osobito za primjene gdje je važna niska potrošnja energije i obrada u stvarnom vremenu. Njihov puni potencijal može se ostvariti samo ako se zajedno unaprijede metode učenja i neuromorfni hardver za opisane modele neurona. Budući radovi trebali bi se usmjeriti na njihovo povezivanje kako bi se ova tehnologija mogla uspješno primijeniti u stvarnim sustavima umjetne inteligencije.

Literatura

- [1] T. Wu, Y. Wang, and N. Quach, “Advancements in Natural Language Processing: Exploring Transformer-Based Architectures for Text Understanding,” Mar. 26, 2025, *arXiv*: arXiv:2503.20227. doi: 10.48550/arXiv.2503.20227.
- [2] K. Alomar, H. I. Aysel, and X. Cai, “RNNs, CNNs and Transformers in Human Action Recognition: A Survey and a Hybrid Model,” Aug. 15, 2024, *arXiv*: arXiv:2407.06162. doi: 10.48550/arXiv.2407.06162.
- [3] Y. O. Sharrab, H. Attar, M. A. H. Eljinini, Y. Al-Omary, and W. E. Al-Momani, “Advancements in Speech Recognition: A Systematic Review of Deep Learning Transformer Models, Trends, Innovations, and Future Directions,” *IEEE Access*, vol. 13, pp. 46925–46940, 2025, doi: 10.1109/ACCESS.2025.3550855.
- [4] K. Roy, A. Jaiswal, and P. Panda, “Towards spike-based machine intelligence with neuromorphic computing,” *Nature*, vol. 575, no. 7784, pp. 607–617, Nov. 2019, doi: 10.1038/s41586-019-1677-2.
- [5] W. Maass, “Networks of spiking neurons: The third generation of neural network models,” *Neural Netw.*, vol. 10, no. 9, pp. 1659–1671, Dec. 1997, doi: 10.1016/S0893-6080(97)00011-7.
- [6] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015, doi: 10.1038/nature14539.
- [7] E. Strubell, A. Ganesh, and A. McCallum, “Energy and Policy Considerations for Deep Learning in NLP,” in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, A. Korhonen, D. Traum, and L. Márquez, Eds., Florence, Italy: Association for Computational Linguistics, Jul. 2019, pp. 3645–3650. doi: 10.18653/v1/P19-1355.
- [8] S. Aquino-Brítez *et al.*, “Towards an Energy Consumption Index for Deep Learning Models: A Comparative Analysis of Architectures, GPUs, and Measurement Tools,” *Sensors*, vol. 25, no. 3, Jan. 2025, doi: 10.3390/s25030846.
- [9] Z. Ji and M. Jiang, “A systematic review of electricity demand for large language models: evaluations, challenges, and solutions,” *Renew. Sustain. Energy Rev.*, vol. 225, p. 116159, Jan. 2026, doi: 10.1016/j.rser.2025.116159.
- [10] L. Kováč, “The 20 W sleep-walkers,” *EMBO Rep.*, vol. 11, no. 1, p. 2, Jan. 2010, doi: 10.1038/embor.2009.266.
- [11] L. R. Iyer and A. Basu, “Unsupervised learning of event-based image recordings using spike-timing-dependent plasticity,” in *2017 International Joint Conference on Neural Networks (IJCNN)*, May 2017, pp. 1840–1846. doi: 10.1109/IJCNN.2017.7966074.
- [12] S. G. Aribe, “Spiking Neural Networks: The Future of Brain-Inspired Computing,” *Int. J. Eng. Trends Technol.*, vol. 73, no. 10, Oct. 2025, doi: 10.14445/22315381/IJETT-V73I10P104.
- [13] W. S. McCulloch and W. Pitts, “A logical calculus of the ideas immanent in nervous activity,” *Bull. Math. Biophys.*, vol. 5, no. 4, pp. 115–133, Dec. 1943, doi: 10.1007/BF02478259.
- [14] F. Rosenblatt, “The perceptron: A probabilistic model for information storage and organization in the brain,” *Psychol. Rev.*, vol. 65, no. 6, pp. 386–408, 1958, doi: 10.1037/h0042519.
- [15] G. Cybenko, “Approximation by superpositions of a sigmoidal function,” *Math. Control Signals Syst.*, vol. 2, no. 4, pp. 303–314, Dec. 1989, doi: 10.1007/BF02551274.

- [16] K. Hornik, M. Stinchcombe, and H. White, “Multilayer feedforward networks are universal approximators,” *Neural Netw.*, vol. 2, no. 5, pp. 359–366, Jan. 1989, doi: 10.1016/0893-6080(89)90020-8.
- [17] V. Kunc and J. Kléma, “Three Decades of Activations: A Comprehensive Survey of 400 Activation Functions for Neural Networks,” Feb. 14, 2024, *arXiv*: arXiv:2402.09092. doi: 10.48550/arXiv.2402.09092.
- [18] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, “Learning representations by back-propagating errors,” *Nature*, vol. 323, no. 6088, pp. 533–536, Oct. 1986, doi: 10.1038/323533a0.
- [19] Y. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller, “Efficient BackProp,” in *Neural Networks: Tricks of the Trade*, vol. 1524, G. B. Orr and K.-R. Müller, Eds., in Lecture Notes in Computer Science, vol. 1524., Berlin, Heidelberg: Springer Berlin Heidelberg, 1998, pp. 9–50. doi: 10.1007/3-540-49430-8_2.
- [20] V. Nair and G. E. Hinton, “Rectified linear units improve restricted boltzmann machines,” in *Proceedings of the 27th International Conference on International Conference on Machine Learning*, in ICML’10. Madison, WI, USA: Omnipress, Lipanj 2010, pp. 807–814.
- [21] K. He, X. Zhang, S. Ren, and J. Sun, “Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification,” Feb. 06, 2015, *arXiv*: arXiv:1502.01852. doi: 10.48550/arXiv.1502.01852.
- [22] L. Trottier, P. Giguère, and B. Chaib-draa, “Parametric Exponential Linear Unit for Deep Convolutional Neural Networks,” Jan. 10, 2018, *arXiv*: arXiv:1605.09332. doi: 10.48550/arXiv.1605.09332.
- [23] A. Apicella, F. Donnarumma, F. Isgrò, and R. Prevete, “A survey on modern trainable activation functions,” *Neural Netw.*, vol. 138, pp. 14–32, Jun. 2021, doi: 10.1016/j.neunet.2021.01.026.
- [24] I. J. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville, and Y. Bengio, “Maxout Networks,” Sep. 20, 2013, *arXiv*: arXiv:1302.4389. doi: 10.48550/arXiv.1302.4389.
- [25] R. K. Srivastava, J. Masci, S. Kazerounian, F. Gomez, and J. Schmidhuber, “Compete to compute,” *Adv. Neural Inf. Process. Syst.*, Jan. 2013.
- [26] A. D. Jagtap and G. E. Karniadakis, “HOW IMPORTANT ARE ACTIVATION FUNCTIONS IN REGRESSION AND CLASSIFICATION? A SURVEY, PERFORMANCE COMPARISON, AND FUTURE DIRECTIONS,” *J. Mach. Learn. Model. Comput.*, vol. 4, no. 1, 2023, doi: 10.1615/JMachLearnModelComput.2023047367.
- [27] S. R. Dubey, S. K. Singh, and B. B. Chaudhuri, “Activation Functions in Deep Learning: A Comprehensive Survey and Benchmark,” Jun. 28, 2022, *arXiv*: arXiv:2109.14545. doi: 10.48550/arXiv.2109.14545.
- [28] J. Lederer, “Activation Functions in Artificial Neural Networks: A Systematic Overview,” Jan. 25, 2021, *arXiv*: arXiv:2101.09957. doi: 10.48550/arXiv.2101.09957.
- [29] Y. Bengio, P. Simard, and P. Frasconi, “Learning long-term dependencies with gradient descent is difficult,” *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 157–166, Mar. 1994, doi: 10.1109/72.279181.
- [30] X. Glorot and Y. Bengio, “Understanding the difficulty of training deep feedforward neural networks,” in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, JMLR Workshop and Conference Proceedings, Mar. 2010, pp. 249–256.
- [31] Y. Lecun, Y. Yere, P. Haffner, Y. Rachmad, and L. Bottou, “Gradient-Based Learning Applied to Document Recognition,” *Proc. IEEE*, vol. 86, pp. 2278–2324, Dec. 1998, doi: 10.1109/5.726791.

- [32] P. Chandra and Y. Singh, “An activation function adapting training algorithm for sigmoidal feedforward networks,” *Neurocomput.*, vol. 61, no. C, pp. 429–437, Listopad 2004, doi: 10.1016/j.neucom.2004.04.001.
- [33] A. Njikam and H. Zhao, “A novel activation function for multilayer feed-forward neural networks,” *Appl. Intell.*, vol. 45, Jul. 2016, doi: 10.1007/s10489-015-0744-0.
- [34] B. Xu, R. Huang, and M. Li, “Revise Saturated Activation Functions,” May 02, 2016, *arXiv*: arXiv:1602.05980. doi: 10.48550/arXiv.1602.05980.
- [35] S. Kong and M. Takatsuka, “Hexpo: A vanishing-proof activation function,” in *2017 International Joint Conference on Neural Networks (IJCNN)*, May 2017, pp. 2562–2567. doi: 10.1109/IJCNN.2017.7966168.
- [36] Y. Qin, X. Wang, and Z. Jingqiang, “The Optimized Deep Belief Networks With Improved Logistic Sigmoid Units and Their Application in Fault Diagnosis for Planetary Gearboxes of Wind Turbines,” *IEEE Trans. Ind. Electron.*, vol. PP, pp. 1–1, Jul. 2018, doi: 10.1109/TIE.2018.2856205.
- [37] S. Elfving, E. Uchibe, and K. Doya, “Sigmoid-Weighted Linear Units for Neural Network Function Approximation in Reinforcement Learning,” Nov. 02, 2017, *arXiv*: arXiv:1702.03118. doi: 10.48550/arXiv.1702.03118.
- [38] S. K. Roy, S. Manna, S. R. Dubey, and B. B. Chaudhuri, “LiSHT: Non-Parametric Linearly Scaled Hyperbolic Tangent Activation Function for Neural Networks,” Feb. 17, 2023, *arXiv*: arXiv:1901.05894. doi: 10.48550/arXiv.1901.05894.
- [39] A. Farzad, H. Mashayekhi, and H. Hassanpour, “A comparative performance analysis of different activation functions in LSTM networks for classification,” *Neural Comput. Appl.*, vol. 31, no. 7, pp. 2507–2521, Jul. 2019, doi: 10.1007/s00521-017-3210-6.
- [40] Y. Zhou, D. Li, S. Huo, and S.-Y. Kung, “Soft-Root-Sign Activation Function,” Mar. 01, 2020, *arXiv*: arXiv:2003.00547. doi: 10.48550/arXiv.2003.00547.
- [41] S. Mastromichalakis, “Parametric Leaky Tanh: A New Hybrid Activation Function for Deep Learning,” Aug. 11, 2023, *arXiv*: arXiv:2310.07720. doi: 10.48550/arXiv.2310.07720.
- [42] S. Kavun, “Hybrid activation functions for deep neural networks: S3 and S4 -- a novel approach to gradient flow optimization,” Jul. 29, 2025, *arXiv*: arXiv:2507.22090. doi: 10.48550/arXiv.2507.22090.
- [43] I. Goodfellow, A. Courville, and Y. Bengio, *Deep learning*. in Adaptive computation and machine learning. Cambridge, Massachusetts: The MIT Press, 2016.
- [44] A. L. Maas, “Rectifier Nonlinearities Improve Neural Network Acoustic Models,” 2013.
- [45] K. He, X. Zhang, S. Ren, and J. Sun, “Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification,” in *2015 IEEE International Conference on Computer Vision (ICCV)*, Dec. 2015, pp. 1026–1034. doi: 10.1109/ICCV.2015.123.
- [46] B. Xu, N. Wang, T. Chen, and M. Li, “Empirical Evaluation of Rectified Activations in Convolutional Network,” Nov. 27, 2015, *arXiv*: arXiv:1505.00853. doi: 10.48550/arXiv.1505.00853.
- [47] W. Shang, K. Sohn, D. Almeida, and H. Lee, “Understanding and Improving Convolutional Neural Networks via Concatenated Rectified Linear Units,” Jul. 19, 2016, *arXiv*: arXiv:1603.05201. doi: 10.48550/arXiv.1603.05201.
- [48] S. S. Liew, M. Khalil-Hani, and R. Bakhteri, “Bounded activation functions for enhanced training stability of deep neural networks on visual pattern recognition problems,” *Neurocomput.*, vol. 216, no. C, pp. 718–734, Prosinac 2016, doi: 10.1016/j.neucom.2016.08.037.
- [49] A. Gupta and R. Duggal, *P-TELU: Parametric Tan Hyperbolic Linear Unit Activation for Deep Neural Networks*. 2017, p. 978. doi: 10.1109/ICCVW.2017.119.

- [50] S. Qiu, X. Xu, and B. Cai, “FReLU: Flexible Rectified Linear Units for Improving Convolutional Neural Networks,” Jan. 29, 2018, *arXiv*: arXiv:1706.08098. doi: 10.48550/arXiv.1706.08098.
- [51] X. Jiang, Y. Pang, X. Li, J. Pan, and Y. Xie, “Deep neural networks with Elastic Rectified Linear Units for object recognition,” *Neurocomputing*, vol. 275, pp. 1132–1139, Jan. 2018, doi: 10.1016/j.neucom.2017.09.056.
- [52] J. Cao, Y. Pang, X. Li, and J. Liang, “Randomly translational activation inspired by the input distributions of ReLU,” *Neurocomputing*, vol. 275, pp. 859–868, Jan. 2018, doi: 10.1016/j.neucom.2017.09.031.
- [53] F. Godin, J. Degraeve, J. Dambre, and W. D. Neve, “Dual Rectified Linear Units (DReLUs): A Replacement for Tanh Activation Functions in Quasi-Recurrent Neural Networks,” *Pattern Recognit. Lett.*, vol. 116, pp. 8–14, Dec. 2018, doi: 10.1016/j.patrec.2018.09.006.
- [54] Z. Tang, L. Luo, H. Peng, and S. Li, “A joint residual network with paired ReLUs activation for image super-resolution,” *Neurocomputing*, vol. 273, pp. 37–46, Jan. 2018, doi: 10.1016/j.neucom.2017.07.061.
- [55] S. R. Dubey and S. Chakraborty, “Average Biased ReLU Based CNN Descriptor for Improved Face Retrieval,” *Multimed. Tools Appl.*, vol. 80, no. 15, pp. 23181–23206, Jun. 2021, doi: 10.1007/s11042-020-10269-x.
- [56] H. Hu, “vReLU Activation Functions for Artificial Neural Networks,” *2018 14th Int. Conf. Nat. Comput. Fuzzy Syst. Knowl. Discov. ICNC-FSKD*, pp. 856–860, Jul. 2018, doi: 10.1109/FSKD.2018.8687140.
- [57] A. Nicolae, “PLU: The Piecewise Linear Unit Activation Function,” Sep. 03, 2018, *arXiv*: arXiv:1809.09534. doi: 10.48550/arXiv.1809.09534.
- [58] D. Macêdo, C. Zanchettin, A. Oliveira, and T. Ludermir, “Enhancing Batch Normalized Convolutional Networks using Displaced Rectifier Linear Units: A Systematic Comparative Study,” *Expert Syst. Appl.*, vol. 124, Jun. 2019, doi: 10.1016/j.eswa.2019.01.066.
- [59] L. B. Godfrey, “An Evaluation of Parametric Activation Functions for Deep Learning,” in *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, Bari, Italy: IEEE Press, Listopad 2019, pp. 3006–3011. doi: 10.1109/SMC.2019.8913972.
- [60] Y. Liu, J. Zhang, C. Gao, J. Qu, and L. Ji, “Natural-Logarithm-Rectified Activation Function in Convolutional Neural Networks,” Aug. 25, 2019, *arXiv*: arXiv:1908.03682. doi: 10.48550/arXiv.1908.03682.
- [61] S. Gu, R. Timofte, and L. V. Gool, “Multi-bin Trainable Linear Unit for Fast Image Restoration Networks,” Jul. 30, 2018, *arXiv*: arXiv:1807.11389. doi: 10.48550/arXiv.1807.11389.
- [62] M. Basirat and P. M. Roth, “L*ReLU: Piece-wise Linear Activation Functions for Deep Fine-grained Visual Categorization,” Oct. 27, 2019, *arXiv*: arXiv:1910.12259. doi: 10.48550/arXiv.1910.12259.
- [63] M. Roodschild, J. Gotay-Sardiñas, V. A. Jimenez, and A. Will, “Zorro: A Flexible and Differentiable Parametric Family of Activation Functions That Extends ReLU and GELU,” Sep. 28, 2024, *arXiv*: arXiv:2409.19239. doi: 10.48550/arXiv.2409.19239.
- [64] J. S. Bridle, “Training stochastic model recognition algorithms as networks can lead to maximum mutual information estimation of parameters,” in *Proceedings of the 3rd International Conference on Neural Information Processing Systems*, in NIPS’89. Cambridge, MA, USA: MIT Press, Siječanj 1989, pp. 211–217.
- [65] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, “Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs),” Feb. 22, 2016, *arXiv*: arXiv:1511.07289. doi: 10.48550/arXiv.1511.07289.

- [66] G. Klambauer, T. Unterthiner, A. Mayr, and S. Hochreiter, “Self-Normalizing Neural Networks,” Sep. 07, 2017, *arXiv*: arXiv:1706.02515. doi: 10.48550/arXiv.1706.02515.
- [67] J. T. Barron, “Continuously Differentiable Exponential Linear Units,” Apr. 24, 2017, *arXiv*: arXiv:1704.07483. doi: 10.48550/arXiv.1704.07483.
- [68] Y. Li, C. Fan, Y. Li, Q. Wu, and Y. Ming, “Improving Deep Neural Network with Multiple Parametric Exponential Linear Units,” Jan. 17, 2017, *arXiv*: arXiv:1606.00305. doi: 10.48550/arXiv.1606.00305.
- [69] M. Basirat and P. M. Roth, “The Quest for the Golden Activation Function,” Aug. 02, 2018, *arXiv*: arXiv:1808.00783. doi: 10.48550/arXiv.1808.00783.
- [70] Y. Ying, J. Su, P. Shan, L. Miao, X. Wang, and S. Peng, “Rectified Exponential Units for Convolutional Neural Networks,” *IEEE Access*, vol. 7, pp. 101633–101640, 2019, doi: 10.1109/ACCESS.2019.2928442.
- [71] Z. Qiumei, T. Dan, and W. Fenghua, “Improved Convolutional Neural Network Based on Fast Exponentially Linear Unit Activation Function,” *IEEE Access*, vol. 7, pp. 151359–151367, 2019, doi: 10.1109/ACCESS.2019.2948112.
- [72] D. Kim, J. Kim, and J. Kim, “Elastic exponential linear units for convolutional neural networks,” *Neurocomputing*, vol. 406, pp. 253–266, Sep. 2020, doi: 10.1016/j.neucom.2020.03.051.
- [73] Q. Cheng, H. Li, Q. Wu, L. Ma, and K. N. Ngan, “Parametric Deformable Exponential Linear Units for deep neural networks,” *Neural Netw. Off. J. Int. Neural Netw. Soc.*, vol. 125, pp. 281–289, May 2020, doi: 10.1016/j.neunet.2020.02.012.
- [74] I. Das, M. Safari, S. Adriaensen, and F. Hutter, “Gompertz Linear Units: Leveraging Asymmetry for Enhanced Learning Dynamics,” May 21, 2025, *arXiv*: arXiv:2502.03654. doi: 10.48550/arXiv.2502.03654.
- [75] A. H. Huang and I. Schlag, “Deriving Activation Functions Using Integration,” Jan. 31, 2025, *arXiv*: arXiv:2411.13010. doi: 10.48550/arXiv.2411.13010.
- [76] F. Agostinelli, M. Hoffman, P. Sadowski, and P. Baldi, “Learning Activation Functions to Improve Deep Neural Networks,” Apr. 21, 2015, *arXiv*: arXiv:1412.6830. doi: 10.48550/arXiv.1412.6830.
- [77] S. Scardapane, M. Scarpiniti, D. Comminiello, and A. Uncini, “Learning activation functions from data using cubic spline interpolation,” vol. 102, 2019, pp. 73–83. doi: 10.1007/978-3-319-95098-3_7.
- [78] A. Mishra, P. Chandra, U. Ghose, and S. Sodhi, “Bi-Modal Derivative Adaptive Activation Function Sigmoidal Feedforward Artificial Neural Networks,” *Appl. Soft Comput.*, vol. 61, Sep. 2017, doi: 10.1016/j.asoc.2017.09.002.
- [79] S. Qian, H. Liu, C. Liu, S. Wu, and H. S. Wong, “Adaptive activation functions in convolutional neural networks,” *Neurocomput.*, vol. 272, no. C, pp. 204–212, Siječanj 2018, doi: 10.1016/j.neucom.2017.06.070.
- [80] P. Ramachandran, B. Zoph, and Q. V. Le, “Searching for Activation Functions,” Oct. 27, 2017, *arXiv*: arXiv:1710.05941. doi: 10.48550/arXiv.1710.05941.
- [81] E. Alcaide, “E-swish: Adjusting Activations to Different Network Depths,” Jan. 22, 2018, *arXiv*: arXiv:1801.07145. doi: 10.48550/arXiv.1801.07145.
- [82] H. H. Chieng, N. Wahid, P. Ong, and S. R. K. Perla, “Flatten-T Swish: a thresholded ReLU-Swish-like activation function for deep learning,” Dec. 15, 2018. doi: 10.26555/ijain.v4i2.249.
- [83] N. Patwardhan, M. Ingahalikar, and R. Walambe, “ARiA: Utilizing Richard’s Curve for Controlling the Non-monotonicity of the Activation Function in Deep Neural Nets,” May 22, 2018, *arXiv*: arXiv:1805.08878. doi: 10.48550/arXiv.1805.08878.

- [84] Ö. F. Ertuğrul, “A novel type of activation function in artificial neural networks: Trained activation function,” *Neural Netw. Off. J. Int. Neural Netw. Soc.*, vol. 99, pp. 148–157, Mar. 2018, doi: 10.1016/j.neunet.2018.01.007.
- [85] M. Goyal, R. Goyal, and B. Lall, “Learning Activation Functions: A new paradigm for understanding Neural Networks,” Dec. 09, 2020, *arXiv*: arXiv:1906.09529. doi: 10.48550/arXiv.1906.09529.
- [86] G. Maguolo, L. Nanni, and S. Ghidoni, “Ensemble of Convolutional Neural Networks Trained with Different Activation Functions,” Sep. 21, 2020, *arXiv*: arXiv:1905.02473. doi: 10.48550/arXiv.1905.02473.
- [87] A. Rajanand and P. Singh, “ErfReLU: Adaptive Activation Function for Deep Neural Network,” Jun. 02, 2023, *arXiv*: arXiv:2306.01822. doi: 10.48550/arXiv.2306.01822.
- [88] X. Wu, Q. Tao, and S. Wang, “RepAct: The Re-parameterizable Adaptive Activation Function,” Jun. 28, 2024, *arXiv*: arXiv:2407.00131. doi: 10.48550/arXiv.2407.00131.
- [89] W. Gerstner and W. M. Kistler, *Spiking Neuron Models: Single Neurons, Populations, Plasticity*, 1st ed. Cambridge University Press, 2002. doi: 10.1017/CBO9780511815706.
- [90] X. Wang, X. Lin, and X. Dang, “Supervised learning in spiking neural networks: A review of algorithms and evaluations,” *Neural Netw.*, vol. 125, pp. 258–280, May 2020, doi: 10.1016/j.neunet.2020.02.011.
- [91] P. Pietrzak, S. Szczyński, D. Huderek, and Ł. Przyborowski, “Overview of Spiking Neural Network Learning Approaches and Their Computational Complexities,” *Sensors*, vol. 23, no. 6, p. 3037, Mar. 2023, doi: 10.3390/s23063037.
- [92] C. Mead, “Neuromorphic electronic systems,” *Proc. IEEE*, vol. 78, no. 10, pp. 1629–1636, Oct. 1990, doi: 10.1109/5.58356.
- [93] S. Laughlin, “Energy as a constraint on the coding and processing of sensory information,” *Curr. Opin. Neurobiol.*, vol. 11, no. 4, pp. 475–480, Aug. 2001, doi: 10.1016/S0959-4388(00)00237-3.
- [94] G. Q. Bi and M. M. Poo, “Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type,” *J. Neurosci. Off. J. Soc. Neurosci.*, vol. 18, no. 24, pp. 10464–10472, Dec. 1998, doi: 10.1523/JNEUROSCI.18-24-10464.1998.
- [95] E. M. Izhikevich, “Which Model to Use for Cortical Spiking Neurons?,” *IEEE Trans. Neural Netw.*, vol. 15, no. 5, pp. 1063–1070, Sep. 2004, doi: 10.1109/TNN.2004.832719.
- [96] P. Davidovits, Ed., *Physics in biology and medicine*, 3rd ed. in Complementary science series. Amsterdam Boston: Elsevier/Academic Press, 2008.
- [97] C. Molnar and J. Gair, “16.2 How Neurons Communicate,” May 2015, Accessed: Jan. 23, 2026. [Online]. Available: <https://opentextbc.ca/biology/chapter/16-2-how-neurons-communicate/>
- [98] H. Zeng and J. R. Sanes, “Neuronal cell-type classification: challenges, opportunities and the path forward,” *Nat. Rev. Neurosci.*, vol. 18, no. 9, pp. 530–546, Sep. 2017, doi: 10.1038/nrn.2017.85.
- [99] “Intelligence Science,” ScienceDirect. Accessed: Jan. 23, 2026. [Online]. Available: <http://www.sciencedirect.com:5070/book/monograph/9780323853804/intelligence-science>
- [100] “35.1: Neurons and Glial Cells,” Biology LibreTexts. Accessed: Jan. 23, 2026. [Online]. Available: [https://bio.libretexts.org/Bookshelves/Introductory_and_General_Biology/General_Biology_1e_\(OpenStax\)/7%3A_Animal_Structure_and_Function/35%3A_The_Nervous_System/35.1%3A_Neurons_and_Glial_Cells](https://bio.libretexts.org/Bookshelves/Introductory_and_General_Biology/General_Biology_1e_(OpenStax)/7%3A_Animal_Structure_and_Function/35%3A_The_Nervous_System/35.1%3A_Neurons_and_Glial_Cells)
- [101] J. C. Eccles, *The physiology of synapses*. in The physiology of synapses. Oxford, England: Academic Press, 1964. doi: 10.1007/978-3-642-64950-9.

- [102] “Types of neurons.” Accessed: Jan. 23, 2026. [Online]. Available: <https://qbi.uq.edu.au/brain/brain-anatomy/types-neurons>
- [103] D. Warland, F. Rieke, R. de R. van Steveninck, and W. Bialek, *Spikes: Exploring the Neural Code*. in Computational Neuroscience Series. Cambridge, MA, USA: MIT Press, 1996.
- [104] D. Auge, J. Hille, E. Mueller, and A. Knoll, “A Survey of Encoding Techniques for Signal Processing in Spiking Neural Networks,” *Neural Process. Lett.*, vol. 53, no. 6, pp. 4693–4710, Dec. 2021, doi: 10.1007/s11063-021-10562-2.
- [105] W. Gerstner, W. M. Kistler, R. Naud, and L. Paninski, *Neuronal Dynamics: From Single Neurons to Networks and Models of Cognition*, 1st ed. Cambridge University Press, 2014. doi: 10.1017/CBO9781107447615.
- [106] F. Ponulak and A. Kasinski, “Ponulak, Filip, and Andrzej Kasinski. ‘Introduction to Spiking Neural Networks: Information Processing, Learning and Applications.’ *Acta Neurobiologiae Experimentalis* 71, no. 4 (2011): 409–33. <https://doi.org/10.55782/ane-2011-1862>,” *Acta Neurobiol. Exp. (Warsz.)*, vol. 71, no. 4, pp. 409–433, Dec. 2011, doi: 10.55782/ane-2011-1862.
- [107] J. J. Hopfield, “Pattern recognition computation using action potential timing for stimulus representation,” *Nature*, vol. 376, no. 6535, pp. 33–36, Jul. 1995, doi: 10.1038/376033a0.
- [108] C. Kayser, M. A. Montemurro, N. K. Logothetis, and S. Panzeri, “Spike-Phase Coding Boosts and Stabilizes Information Carried by Spatial and Temporal Spike Patterns,” *Neuron*, vol. 61, no. 4, pp. 597–608, Feb. 2009, doi: 10.1016/j.neuron.2009.01.008.
- [109] D. L. Manna, A. Vicente-Sola, P. Kirkland, T. J. Bihl, and G. Di Caterina, “Simple and complex spiking neurons: perspectives and analysis in a simple STDP scenario,” *Neuromorphic Comput. Eng.*, vol. 2, no. 4, p. 044009, Dec. 2022, doi: 10.1088/2634-4386/ac999b.
- [110] A. L. Hodgkin and A. F. Huxley, “A quantitative description of membrane current and its application to conduction and excitation in nerve,” *J. Physiol.*, vol. 117, no. 4, pp. 500–544, Aug. 1952, doi: 10.1113/jphysiol.1952.sp004764.
- [111] R. FitzHugh, “Impulses and Physiological States in Theoretical Models of Nerve Membrane,” *Biophys. J.*, vol. 1, no. 6, pp. 445–466, Jul. 1961, doi: 10.1016/S0006-3495(61)86902-6.
- [112] J. L. Hindmarsh and R. M. Rose, “A model of neuronal bursting using three coupled first order differential equations,” *Proc. R. Soc. Lond. B Biol. Sci.*, vol. 221, no. 1222, pp. 87–102, Mar. 1984, doi: 10.1098/rspb.1984.0024.
- [113] C. Morris and H. Lecar, “Voltage oscillations in the barnacle giant muscle fiber,” *Biophys. J.*, vol. 35, no. 1, pp. 193–213, Jul. 1981, doi: 10.1016/S0006-3495(81)84782-0.
- [114] H. R. Wilson and J. D. Cowan, “Excitatory and inhibitory interactions in localized populations of model neurons,” *Biophys. J.*, vol. 12, no. 1, pp. 1–24, Jan. 1972, doi: 10.1016/S0006-3495(72)86068-5.
- [115] H. R. Wilson and J. D. Cowan, “A mathematical theory of the functional dynamics of cortical and thalamic nervous tissue,” *Kybernetik*, vol. 13, no. 2, pp. 55–80, Sep. 1973, doi: 10.1007/BF00288786.
- [116] G. L. Gerstein and B. Mandelbrot, “RANDOM WALK MODELS FOR THE SPIKE ACTIVITY OF A SINGLE NEURON,” *Biophys. J.*, vol. 4, no. 1 Pt 1, pp. 41–68, Jan. 1964, doi: 10.1016/s0006-3495(64)86768-0.
- [117] A. N. Burkitt, “A Review of the Integrate-and-fire Neuron Model: I. Homogeneous Synaptic Input,” *Biol. Cybern.*, vol. 95, no. 1, pp. 1–19, Jul. 2006, doi: 10.1007/s00422-006-0068-6.

- [118] D. Tal and E. L. Schwartz, “Computing with the leaky integrate-and-fire neuron: logarithmic computation and multiplication,” *Neural Comput.*, vol. 9, no. 2, pp. 305–318, Feb. 1997, doi: 10.1162/neco.1997.9.2.305.
- [119] “Methods in Neuronal Modeling: From Ions to Networks | Guide books | ACM Digital Library,” Guide books. Accessed: Jan. 28, 2026. [Online]. Available: <https://dl.acm.org/doi/book/10.5555/551736>
- [120] N. Fourcaud-Trocmé, D. Hansel, C. van Vreeswijk, and N. Brunel, “How spike generation mechanisms determine the neuronal response to fluctuating inputs,” *J. Neurosci. Off. J. Soc. Neurosci.*, vol. 23, no. 37, pp. 11628–11640, Dec. 2003, doi: 10.1523/JNEUROSCI.23-37-11628.2003.
- [121] R. Brette and W. Gerstner, “Adaptive Exponential Integrate-and-Fire Model as an Effective Description of Neuronal Activity,” *J. Neurophysiol.*, vol. 94, no. 5, pp. 3637–3642, Nov. 2005, doi: 10.1152/jn.00686.2005.
- [122] S. Haghiri and A. Ahmadi, “A Novel Digital Realization of AdEx Neuron Model,” *IEEE Trans. Circuits Syst. II Express Briefs*, vol. 67, no. 8, pp. 1444–1448, Aug. 2020, doi: 10.1109/TCSII.2019.2938180.
- [123] E. M. Izhikevich, “Resonate-and-fire neurons,” *Neural Netw.*, vol. 14, no. 6, pp. 883–894, Jul. 2001, doi: 10.1016/S0893-6080(01)00078-8.
- [124] W. Teka, T. M. Marinov, and F. Santamaria, “Neuronal Spike Timing Adaptation Described with a Fractional Leaky Integrate-and-Fire Model,” *PLoS Comput. Biol.*, vol. 10, no. 3, p. e1003526, Mar. 2014, doi: 10.1371/journal.pcbi.1003526.
- [125] G. Bellec, D. Salaj, A. Subramoney, R. Legenstein, and W. Maass, “Long short-term memory and learning-to-learn in networks of spiking neurons,” Dec. 25, 2018, *arXiv*: arXiv:1803.09574. doi: 10.48550/arXiv.1803.09574.
- [126] S. Higuchi, S. Kairat, S. M. Bohte, and S. Otte, “Balanced Resonate-and-Fire Neurons,” Oct. 08, 2024, *arXiv*: arXiv:2402.14603. doi: 10.48550/arXiv.2402.14603.
- [127] E. M. Izhikevich, “Simple model of spiking neurons,” *IEEE Trans. Neural Netw.*, vol. 14, no. 6, pp. 1569–1572, Nov. 2003, doi: 10.1109/TNN.2003.820440.
- [128] H. Wang and H. Wang, “Improvement of Izhikevich’s Neuronal and Neural Network Model,” in *2009 International Conference on Information Engineering and Computer Science*, Dec. 2009, pp. 1–4. doi: 10.1109/ICIECS.2009.5363122.
- [129] C. Börgers, “An Introduction to Modeling Neuronal Dynamics,” *Texts Appl. Math.*, 2017, doi: 10.1007/978-3-319-51171-9.
- [130] S. Thorpe and J. Gautrais, “Rank Order Coding,” in *Computational Neuroscience*, J. M. Bower, Ed., Boston, MA: Springer US, 1998, pp. 113–118.
- [131] J. H. Lee, T. Delbruck, and M. Pfeiffer, “Training Deep Spiking Neural Networks using Backpropagation,” Aug. 31, 2016, *arXiv*: arXiv:1608.08782. doi: 10.48550/arXiv.1608.08782.
- [132] A. Tavanaei, M. Ghodrati, S. R. Kheradpisheh, T. Masquelier, and A. S. Maida, “Deep Learning in Spiking Neural Networks,” *Neural Netw.*, vol. 111, pp. 47–63, Mar. 2019, doi: 10.1016/j.neunet.2018.12.002.
- [133] S. R. Kheradpisheh, M. Ganjtabesh, S. J. Thorpe, and T. Masquelier, “STDP-based spiking deep convolutional neural networks for object recognition,” *Neural Netw.*, vol. 99, pp. 56–67, Mar. 2018, doi: 10.1016/j.neunet.2017.12.005.
- [134] A. Amir *et al.*, “A Low Power, Fully Event-Based Gesture Recognition System,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 7388–7397. doi: 10.1109/CVPR.2017.781.
- [135] S. B. Laughlin and T. J. Sejnowski, “Communication in Neuronal Networks,” *Science*, vol. 301, no. 5641, pp. 1870–1874, Sep. 2003, doi: 10.1126/science.1089662.

- [136] X. Pitkow and M. Meister, “Decorrelation and efficient coding by retinal ganglion cells,” *Nat. Neurosci.*, vol. 15, no. 4, pp. 628–635, Apr. 2012, doi: 10.1038/nn.3064.
- [137] N. Lynch, C. Musco, and M. Parter, “Winner-Take-All Computation in Spiking Neural Networks,” Apr. 25, 2019, *arXiv*: arXiv:1904.12591. doi: 10.48550/arXiv.1904.12591.
- [138] Y. Dong, D. Zhao, Y. Li, and Y. Zeng, “An Unsupervised STDP-based Spiking Neural Network Inspired By Biologically Plausible Learning Rules and Connections,” Apr. 22, 2023, *arXiv*: arXiv:2207.02727. doi: 10.48550/arXiv.2207.02727.
- [139] Y. Cao, Y. Chen, and D. Khosla, “Spiking Deep Convolutional Neural Networks for Energy-Efficient Object Recognition,” *Int. J. Comput. Vis.*, vol. 113, no. 1, pp. 54–66, May 2015, doi: 10.1007/s11263-014-0788-3.
- [140] E. O. Neftci, H. Mostafa, and F. Zenke, “Surrogate Gradient Learning in Spiking Neural Networks,” May 03, 2019, *arXiv*: arXiv:1901.09948. doi: 10.48550/arXiv.1901.09948.
- [141] M. A. Agebure, P. A. Wumnaya, and E. Y. Baagyere, “A Survey of Supervised Learning Models for Spiking Neural Network,” *Asian J. Res. Comput. Sci.*, pp. 35–49, Jun. 2021, doi: 10.9734/ajrcos/2021/v9i430228.
- [142] S. Bohté, J. Kok, and H. L. Poutre, “SpikeProp: backpropagation for networks of spiking neurons,” presented at the The European Symposium on Artificial Neural Networks, 2000.
- [143] Jianguo Xin and M. J. Embrechts, “Supervised learning with spiking neural networks,” *IJCNN01 Int. Jt. Conf. Neural Netw. Proc. Cat No01CH37222*, vol. 3, pp. 1772–1777, 2001, doi: 10.1109/IJCNN.2001.938430.
- [144] S. McKennoch, Dingding Liu, and L. G. Bushnell, “Fast Modifications of the SpikeProp Algorithm,” in *The 2006 IEEE International Joint Conference on Neural Network Proceedings*, Vancouver, BC, Canada: IEEE, 2006, pp. 3970–3977. doi: 10.1109/IJCNN.2006.246918.
- [145] F. Zenke and S. Ganguli, “SuperSpike: Supervised Learning in Multilayer Spiking Neural Networks,” *Neural Comput.*, vol. 30, no. 6, pp. 1514–1541, Jun. 2018, doi: 10.1162/neco_a_01086.
- [146] S. B. Shrestha and G. Orchard, “SLAYER: Spike Layer Error Reassignment in Time,” 2018, *arXiv*. doi: 10.48550/ARXIV.1810.08646.
- [147] T. C. Wunderlich and C. Pehle, “Event-based backpropagation can compute exact gradients for spiking neural networks,” *Sci. Rep.*, vol. 11, no. 1, p. 12829, Jun. 2021, doi: 10.1038/s41598-021-91786-z.
- [148] D. O. Hebb, *The Organization of Behavior*, 0 ed. Psychology Press, 2005. doi: 10.4324/9781410612403.
- [149] H. Z. Shouval, “Models of synaptic plasticity,” *Scholarpedia*, vol. 2, no. 7, p. 1605, Jul. 2007, doi: 10.4249/scholarpedia.1605.
- [150] R. Kempter, W. Gerstner, and J. L. van Hemmen, “Hebbian learning and spiking neurons,” *Phys. Rev. E*, vol. 59, no. 4, pp. 4498–4514, Apr. 1999, doi: 10.1103/PhysRevE.59.4498.
- [151] P. Sjöström and W. Gerstner, “Spike-timing dependent plasticity,” *Scholarpedia*, vol. 5, p. 1362, Jan. 2010, doi: 10.4249/scholarpedia.1362.
- [152] N. Frémaux and W. Gerstner, “Neuromodulated Spike-Timing-Dependent Plasticity, and Theory of Three-Factor Learning Rules,” *Front. Neural Circuits*, vol. 9, Jan. 2016, doi: 10.3389/fncir.2015.00085.
- [153] S. Song, K. D. Miller, and L. F. Abbott, “Competitive Hebbian learning through spike-timing-dependent synaptic plasticity,” *Nat. Neurosci.*, vol. 3, no. 9, pp. 919–926, Sep. 2000, doi: 10.1038/78829.
- [154] K. Yamazaki, V.-K. Vo-Ho, D. Bulsara, and N. Le, “Spiking Neural Networks and Their Applications: A Review,” *Brain Sci.*, vol. 12, no. 7, p. 863, Jun. 2022, doi: 10.3390/brainsci12070863.

- [155] N. Caporale and Y. Dan, “Spike Timing–Dependent Plasticity: A Hebbian Learning Rule,” *Annu. Rev. Neurosci.*, vol. 31, no. Volume 31, 2008, pp. 25–46, Jul. 2008, doi: 10.1146/annurev.neuro.31.060407.125639.
- [156] F. Paredes-Vallés, K. Y. W. Scheper, and G. C. H. E. de Croon, “Unsupervised Learning of a Hierarchical Spiking Neural Network for Optical Flow Estimation: From Events to Global Motion Perception,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 8, pp. 2051–2064, Aug. 2020, doi: 10.1109/TPAMI.2019.2903179.
- [157] A. Gautam and T. Kohno, “Adaptive STDP-based on-chip spike pattern detection,” *Front. Neurosci.*, vol. 17, p. 1203956, Jul. 2023, doi: 10.3389/fnins.2023.1203956.
- [158] E. Oja, “Oja learning rule,” *Scholarpedia*, vol. 3, no. 3, p. 3612, Mar. 2008, doi: 10.4249/scholarpedia.3612.
- [159] B. S. Blais and L. Cooper, “BCM theory,” *Scholarpedia*, vol. 3, no. 3, p. 1570, Mar. 2008, doi: 10.4249/scholarpedia.1570.
- [160] J. M. Cortes, M. Desroches, S. Rodrigues, R. Veltz, M. A. Muñoz, and T. J. Sejnowski, “Short-term synaptic plasticity in the deterministic Tsodyks-Markram model leads to unpredictable network dynamics,” *Proc. Natl. Acad. Sci. U. S. A.*, vol. 110, no. 41, pp. 16610–16615, Oct. 2013, doi: 10.1073/pnas.1316071110.
- [161] N. Frémaux and W. Gerstner, “Neuromodulated Spike-Timing-Dependent Plasticity, and Theory of Three-Factor Learning Rules,” *Front. Neural Circuits*, vol. 9, p. 85, 2015, doi: 10.3389/fncir.2015.00085.
- [162] E. M. Izhikevich, “Solving the distal reward problem through linkage of STDP and dopamine signaling,” *Cereb. Cortex*, vol. 17, no. 10, pp. 2443–2452, Oct. 2007, doi: 10.1093/cercor/bhl152.
- [163] M. A. Farries and A. L. Fairhall, “Reinforcement learning with modulated spike timing dependent synaptic plasticity,” *J. Neurophysiol.*, vol. 98, no. 6, pp. 3648–3665, Dec. 2007, doi: 10.1152/jn.00364.2007.
- [164] N. Frémaux, H. Sprekeler, and W. Gerstner, “Reinforcement Learning Using a Continuous Time Actor-Critic Framework with Spiking Neurons,” *PLOS Comput. Biol.*, vol. 9, no. 4, p. e1003024, tra 2013, doi: 10.1371/journal.pcbi.1003024.
- [165] F. Ponulak, “ReSuMe-New Supervised Learning Method for Spiking Neural Networks,” 2005.
- [166] G. Bellec, F. Scherr, E. Hajek, D. Salaj, R. Legenstein, and W. Maass, “Biologically inspired alternatives to backpropagation through time for learning in recurrent neural nets,” Feb. 21, 2019, *arXiv*: arXiv:1901.09049. doi: 10.48550/arXiv.1901.09049.
- [167] C. S. Thakur *et al.*, “Large-Scale Neuromorphic Spiking Array Processors: A quest to mimic the brain,” May 23, 2018, *arXiv*: arXiv:1805.08932. doi: 10.48550/arXiv.1805.08932.
- [168] N. Rathi *et al.*, “Exploring Neuromorphic Computing Based on Spiking Neural Networks: Algorithms to Hardware,” *ACM Comput Surv*, vol. 55, no. 12, p. 243:1-243:49, Ožujak 2023, doi: 10.1145/3571155.
- [169] M. V. DeBole *et al.*, “TrueNorth: Accelerating From Zero to 64 Million Neurons in 10 Years,” *Computer*, vol. 52, no. 5, pp. 20–29, May 2019, doi: 10.1109/MC.2019.2903009.
- [170] M. Davies *et al.*, “Loihi: A Neuromorphic Manycore Processor with On-Chip Learning,” *IEEE Micro*, vol. 38, no. 1, pp. 82–99, Jan. 2018, doi: 10.1109/MM.2018.112130359.
- [171] S. B. Furber *et al.*, “Overview of the SpiNNaker System Architecture,” *IEEE Trans. Comput.*, vol. 62, no. 12, pp. 2454–2467, Dec. 2013, doi: 10.1109/TC.2012.142.
- [172] J. Shen *et al.*, “Darwin: a neuromorphic hardware co-processor based on Spiking Neural Networks,” *Sci. China Inf. Sci.*, vol. 59, no. 2, pp. 1–5, Feb. 2016, doi: 10.1007/s11432-015-5511-7.

- [173] B. V. Benjamin *et al.*, “Neurogrid: A Mixed-Analog-Digital Multichip System for Large-Scale Neural Simulations,” *Proc. IEEE*, vol. 102, no. 5, pp. 699–716, May 2014, doi: 10.1109/JPROC.2014.2313565.
- [174] A. Neckar *et al.*, “Braindrop: A Mixed-Signal Neuromorphic Architecture With a Dynamical Systems-Based Programming Model,” *Proc. IEEE*, vol. 107, no. 1, pp. 144–164, Jan. 2019, doi: 10.1109/JPROC.2018.2881432.
- [175] A. Grübl, S. Billaudelle, B. Cramer, V. Karasenko, and J. Schemmel, “Verification and Design Methods for the BrainScaleS Neuromorphic Hardware System,” *J. Signal Process. Syst.*, vol. 92, no. 11, pp. 1277–1292, Nov. 2020, doi: 10.1007/s11265-020-01558-7.
- [176] “How neuromorphic computing takes inspiration from our brains,” IBM Research. Accessed: Feb. 20, 2026. [Online]. Available: <https://research.ibm.com/blog/what-is-neuromorphic-or-brain-inspired-computing>
- [177] “An energy-efficient analog chip for AI inference,” IBM Research. Accessed: Feb. 20, 2026. [Online]. Available: https://research.ibm.com/blog/analog-ai-chip-inference?utm_source=chatgpt.com
- [178] “High-performance deep spiking neural networks with 0.3 spikes per neuron for Nature Communications - IBM Research.” Accessed: Feb. 20, 2026. [Online]. Available: <https://research.ibm.com/publications/high-performance-deep-spiking-neural-networks-with-03-spikes-per-neuron>
- [179] T. Carnevale, “Neuron simulation environment,” *Scholarpedia*, vol. 2, no. 6, p. 1378, Jun. 2007, doi: 10.4249/scholarpedia.1378.
- [180] J. M. Bower and D. Beeman, “GENESIS (simulation environment),” *Scholarpedia*, vol. 2, no. 3, p. 1383, Mar. 2007, doi: 10.4249/scholarpedia.1383.
- [181] M.-O. Gewaltig and M. Diesmann, “NEST (NEural Simulation Tool),” *Scholarpedia*, vol. 2, no. 4, p. 1430, Apr. 2007, doi: 10.4249/scholarpedia.1430.
- [182] M. Stimberg, R. Brette, and D. F. Goodman, “Brian 2, an intuitive and efficient neural simulator,” *eLife*, vol. 8, p. e47314, Aug. 2019, doi: 10.7554/eLife.47314.
- [183] A. P. Davison *et al.*, “PyNN: a common interface for neuronal network simulators,” *Front. Neuroinformatics*, vol. 2, Jan. 2009, doi: 10.3389/neuro.11.011.2008.
- [184] W. Fang *et al.*, “SpikingJelly: An open-source machine learning infrastructure platform for spike-based intelligence,” Oct. 25, 2023, *arXiv*: arXiv:2310.16620. doi: 10.48550/arXiv.2310.16620.
- [185] Y. Zeng *et al.*, “BrainCog: A Spiking Neural Network based Brain-inspired Cognitive Intelligence Engine for Brain-inspired AI and Brain Simulation,” *Patterns*, vol. 4, no. 8, p. 100789, Aug. 2023, doi: 10.1016/j.patter.2023.100789.
- [186] A. Paszke *et al.*, “PyTorch: An Imperative Style, High-Performance Deep Learning Library,” Dec. 03, 2019, *arXiv*: arXiv:1912.01703. doi: 10.48550/arXiv.1912.01703.
- [187] C.-G. Pehle and J. Egholm Pedersen, “Norse - A deep learning library for spiking neural networks,” *Zenodo*, Jan. 2021, doi: 10.5281/zenodo.4422025.
- [188] H. Hazan, D. J. Saunders, H. Khan, D. T. Sanghavi, H. T. Siegelmann, and R. Kozma, “BindsNET: A machine learning-oriented spiking neural networks library in Python,” *Front. Neuroinformatics*, vol. 12, p. 89, Dec. 2018, doi: 10.3389/fninf.2018.00089.
- [189] M. Mozafari, M. Ganjtabesh, A. Nowzari-Dalini, and T. Masquelier, “SpykeTorch: Efficient Simulation of Convolutional Spiking Neural Networks with at most one Spike per Neuron,” *Front. Neurosci.*, vol. 13, p. 625, Jul. 2019, doi: 10.3389/fnins.2019.00625.
- [190] E. Yavuz, J. Turner, and T. Nowotny, “GeNN: a code generation framework for accelerated brain simulations,” *Sci. Rep.*, vol. 6, no. 1, p. 18854, Jan. 2016, doi: 10.1038/srep18854.
- [191] B. Rueckauer, I.-A. Lungu, Y. Hu, M. Pfeiffer, and S.-C. Liu, “Conversion of Continuous-Valued Deep Networks to Efficient Event-Driven Networks for Image

- Classification,” *Front. Neurosci.*, vol. 11, p. 682, Dec. 2017, doi: 10.3389/fnins.2017.00682.
- [192] T. Bekolay *et al.*, “Nengo: a Python tool for building large-scale functional brain models,” *Front. Neuroinformatics*, vol. 7, p. 48, Jan. 2014, doi: 10.3389/fninf.2013.00048.
- [193] D. Rasmussen, “NengoDL: Combining deep learning and neuromorphic modelling methods,” Mar. 27, 2019, *arXiv*: arXiv:1805.11144. doi: 10.48550/arXiv.1805.11144.
- [194] *lava-nc/lava*. (Mar. 22, 2026). Jupyter Notebook. Lava. Accessed: Mar. 10, 2026. [Online]. Available: <https://github.com/lava-nc/lava>
- [195] M. J. Page *et al.*, “The PRISMA 2020 statement: an updated guideline for reporting systematic reviews,” *BMJ*, vol. 372, p. n71, Mar. 2021, doi: 10.1136/bmj.n71.
- [196] M. S. Asghar, S. Arslan, and H. Kim, “A Low-Power Spiking Neural Network Chip Based on a Compact LIF Neuron and Binary Exponential Charge Injector Synapse Circuits,” *Sensors*, vol. 21, no. 13, p. 4462, Jun. 2021, doi: 10.3390/s21134462.
- [197] A. Khajooei, M. (Behdad) Jamshidi, and S. B. Shokouhi, “A Super-Efficient TinyML Processor for the Edge Metaverse,” *Information*, vol. 14, no. 4, p. 235, Apr. 2023, doi: 10.3390/info14040235.
- [198] C. Zou, X. Cui, G. Chen, Y. Jiang, and Y. Wang, “Toward a Lossless Conversion for Spiking Neural Networks with Negative-Spike Dynamics,” *Adv. Intell. Syst.*, vol. 5, no. 12, p. 2300383, 2023, doi: 10.1002/aisy.202300383.
- [199] L. Feng, Q. Liu, H. Tang, D. Ma, and G. Pan, “Multi-Level Firing with Spiking DS-ResNet: Enabling Better and Deeper Directly-Trained Spiking Neural Networks,” Apr. 19, 2023, *arXiv*: arXiv:2210.06386. doi: 10.48550/arXiv.2210.06386.
- [200] C. Lee, S. S. Sarwar, P. Panda, G. Srinivasan, and K. Roy, “Enabling Spike-based Backpropagation for Training Deep Neural Network Architectures,” *Front. Neurosci.*, vol. 14, p. 119, Feb. 2020, doi: 10.3389/fnins.2020.00119.
- [201] C. Shi, L. Wang, H. Gao, and M. Tian, “Learnable Leakage and Onset-Spiking Self-Attention in SNNs with Local Error Signals,” *Sensors*, vol. 23, no. 24, p. 9781, Dec. 2023, doi: 10.3390/s23249781.
- [202] S. Dora, S. Sundaram, and N. Sundararajan, “A two stage learning algorithm for a Growing-Pruning Spiking Neural Network for pattern classification problems,” in *2015 International Joint Conference on Neural Networks (IJCNN)*, Killarney, Ireland: IEEE, Jul. 2015, pp. 1–7. doi: 10.1109/IJCNN.2015.7280592.
- [203] V. Saranirad, S. Dora, T. M. McGinnity, and D. Coyle, “CDNA-SNN: A New Spiking Neural Network for Pattern Classification Using Neuronal Assemblies,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 36, no. 2, pp. 2274–2287, Feb. 2025, doi: 10.1109/TNNLS.2024.3353571.
- [204] R. C. Gerum and A. Schilling, “Integration of Leaky-Integrate-and-Fire Neurons in Standard Machine Learning Architectures to Generate Hybrid Networks: A Surrogate Gradient Approach,” *Neural Comput.*, vol. 33, no. 10, pp. 2827–2852, Sep. 2021, doi: 10.1162/neco_a_01424.
- [205] X. Liu, “Inhibition SNN: unveiling the efficacy of various lateral inhibition learning in image pattern recognition,” *Discov. Appl. Sci.*, vol. 6, no. 11, p. 611, Nov. 2024, doi: 10.1007/s42452-024-06332-z.
- [206] B. Loyola-Jara, G. Fernández-Rodríguez, and J. Baladron, “Evolving spiking neural networks: the role of neuron models and encoding schemes in neuromorphic learning,” *Front. Neurosci.*, vol. 20, Feb. 2026, doi: 10.3389/fnins.2026.1697163.
- [207] B. A. Abubaker, J. Razmara, and J. Karimpour, “A Novel Approach for Target Attraction and Obstacle Avoidance of a Mobile Robot in Unknown Environments Using a Customized Spiking Neural Network,” *Appl. Sci.*, vol. 13, no. 24, p. 13145, Dec. 2023, doi: 10.3390/app132413145.

- [208] A. Tavanaei and A. S. Maida, “A Spiking Network that Learns to Extract Spike Signatures from Speech Signals,” *Neurocomputing*, vol. 240, pp. 191–199, May 2017, doi: 10.1016/j.neucom.2017.01.088.
- [209] “Digital Implementation of a Spiking Convolutional Neural Network for Tumor Detection,” *Inf. MIDEM - J. Microelectron. Electron. Compon. Mater.*, pp. 193–202, Jan. 2020, doi: 10.33180/InfMIDEM2019.401.
- [210] “The Lightweight Edge-Side Fault Diagnosis Approach Based on Spiking Neural Network,” *Comput. Mater. Contin.*, vol. 79, no. 3, pp. 4883–4904, Jun. 2024, doi: 10.32604/cmc.2024.051860.
- [211] B. Ayasi, C. J. Carmona, M. Saleh, and A. M. García-Vico, “A Practical Tutorial on Spiking Neural Networks: Comprehensive Review, Models, Experiments, Software Tools, and Implementation Guidelines,” *Eng.*, vol. 6, no. 11, p. 304, Nov. 2025, doi: 10.3390/eng6110304.
- [212] R. Gulakala and M. Stoffel, “A sustainable neuromorphic framework for disease diagnosis using digital medical imaging,” *Comput. Methods Programs Biomed. Update*, vol. 7, p. 100171, 2025, doi: 10.1016/j.cmpbup.2024.100171.
- [213] X. Zhong and H. Pan, “A Spike Neural Network Model for Lateral Suppression of Spike-Timing-Dependent Plasticity with Adaptive Threshold,” *Appl. Sci.*, vol. 12, no. 12, p. 5980, Jun. 2022, doi: 10.3390/app12125980.
- [214] L. Fernando Herbozo Contreras, L. Yu, Z. Huang, Z. Zhang, A. Nikpour, and O. Kavehei, “Tiny dLIF: a dendritic spiking neural network enabling a time-domain energy-efficient seizure detection system,” *Neuromorphic Comput. Eng.*, vol. 5, no. 1, p. 014015, Mar. 2025, doi: 10.1088/2634-4386/adc0b9.
- [215] Y. Garipova, S. Yonekura, and Y. Kuniyoshi, “Noise and Dynamical Synapses as Optimization Tools for Spiking Neural Networks,” *Entropy*, vol. 27, no. 3, p. 219, Mar. 2025, doi: 10.3390/e27030219.
- [216] Z. Rudnicka, J. Szczepanski, and A. Pregowska, “Impact of Neuron Models on Spiking Neural Networks performance. A Complexity Based Classification Approach,” Aug. 24, 2025, *arXiv*: arXiv:2509.06970. doi: 10.48550/arXiv.2509.06970.
- [217] K. Takala, W. Thamviset, and S. Wongthanavas, “BIASNN: a biologically inspired attention mechanism in spiking neural networks for image classification,” *Sci. Rep.*, vol. 15, no. 1, p. 38753, Nov. 2025, doi: 10.1038/s41598-025-22430-3.
- [218] A. Yanguas-Gil, “Coarse scale representation of spiking neural networks: backpropagation through spikes and application to neuromorphic hardware,” Jul. 13, 2020, *arXiv*: arXiv:2007.06176. doi: 10.48550/arXiv.2007.06176.
- [219] G. Yang *et al.*, “Unsupervised Spiking Neural Network with Dynamic Learning of Inhibitory Neurons,” *Sensors*, vol. 23, no. 16, p. 7232, Aug. 2023, doi: 10.3390/s23167232.
- [220] S. Sharmin, N. Rathi, P. Panda, and K. Roy, “Inherent Adversarial Robustness of Deep Spiking Neural Networks: Effects of Discrete Input Encoding and Non-Linear Activations,” Jul. 23, 2020, *arXiv*: arXiv:2003.10399. doi: 10.48550/arXiv.2003.10399.
- [221] E. Stromatias, M. Soto, T. Serrano-Gotarredona, and B. Linares-Barranco, “An Event-Driven Classifier for Spiking Neural Networks Fed with Synthetic or Dynamic Vision Sensor Data,” *Front. Neurosci.*, vol. 11, Jun. 2017, doi: 10.3389/fnins.2017.00350.
- [222] B. Chakraborty and S. Mukhopadhyay, “Heterogeneous Recurrent Spiking Neural Network for Spatio-Temporal Classification,” Sep. 22, 2022, *arXiv*: arXiv:2211.04297. doi: 10.48550/arXiv.2211.04297.
- [223] Y. Li, X. Cui, Y. Zhou, and Y. Li, “A Comparative Study on the Performance and Security Evaluation of Spiking Neural Networks,” *IEEE Access*, vol. 10, pp. 117572–117581, 2022, doi: 10.1109/ACCESS.2022.3220367.

- [224] J. S. Maskeen and S. Lashkare, “A Unified Platform to Evaluate STDP Learning Rule and Synapse Model using Pattern Recognition in a Spiking Neural Network,” arXiv.org. Accessed: Mar. 10, 2026. [Online]. Available: <https://arxiv.org/abs/2506.19377v1>
- [225] M. Kolhar, “A neuromorphic approach to early arrhythmia detection,” *Sci. Rep.*, vol. 15, no. 1, p. 39635, Nov. 2025, doi: 10.1038/s41598-025-23248-9.
- [226] A. Rana and K. K. Kim, “Electrocardiography Classification with Leaky Integrate-and-Fire Neurons in an Artificial Neural Network-Inspired Spiking Neural Network Framework,” *Sensors*, vol. 24, no. 11, p. 3426, May 2024, doi: 10.3390/s24113426.
- [227] Y. Lee, “Abrupt Change Detection of ECG by Spiking Neural Networks: Policy-Aware Operating Points for Edge-Level MI Screening,” *Appl. Sci.*, vol. 15, no. 22, p. 12210, Nov. 2025, doi: 10.3390/app152212210.
- [228] Y. Shi, H. Li, H. Zhang, Z. Wu, and S. Ren, “Accurate and Efficient LIF-Nets for 3D Detection and Recognition,” *IEEE Access*, vol. 8, pp. 98562–98571, 2020, doi: 10.1109/ACCESS.2020.2995886.
- [229] H. Slimi, I. Cherif, S. Abid, and M. Sayadi, “Biologically inspired hybrid model for Alzheimer’s disease classification using structural MRI in the ADNI dataset,” *Front. Artif. Intell.*, vol. 8, Jun. 2025, doi: 10.3389/frai.2025.1590599.
- [230] G. S. Jebaraj and K. Elango, “EEG seizure classification with temporal spiking neural networks and mutual information-based feature selection,” *J. Eng. Appl. Sci.*, vol. 72, no. 1, p. 218, Nov. 2025, doi: 10.1186/s44147-025-00796-5.
- [231] R. Sahu, P. K. Pattnaik, K. S. M. Anbananthen, and S. Muthaiyah, “Identification of Depression Patients Using LIF Spiking Neural Network Model From the Pattern of EEG Signals,” *IEEE Access*, vol. 13, pp. 55156–55168, 2025, doi: 10.1109/ACCESS.2025.3552619.
- [232] C. Zhang, Z. Xiao, and Z. Sheng, “A bearing fault diagnosis method based on a convolutional spiking neural network with spatial–temporal feature-extraction capability,” *Transp. Saf. Environ.*, vol. 5, no. 2, Mar. 2023, doi: 10.1093/tse/tdac050.
- [233] B. Chakraborty, X. She, and S. Mukhopadhyay, “A Fully Spiking Hybrid Neural Network for Energy-Efficient Object Detection,” *IEEE Trans. Image Process.*, vol. 30, pp. 9014–9029, 2021, doi: 10.1109/TIP.2021.3122092.
- [234] S. Xiang, S. Jiang, X. Liu, T. Zhang, and L. Yu, “Spiking VGG7: Deep Convolutional Spiking Neural Network with Direct Training for Object Recognition,” *Electronics*, vol. 11, no. 13, p. 2097, Jan. 2022, doi: 10.3390/electronics11132097.

Popis oznaka i kratica

AER	Address Event Representation
AF	Aktivacijska funkcija
ANN	Artificial Neuron Networks
aSTDP	Adaptive Spike-Timing-Dependent Plasticity
CMOS	Complementary Metal-Oxide-Semiconductor
CNN	Convolutional Neural Networks
CPU	Central processing unit
CT	Computed Tomography
DVS	Dynamic Vision Sensor
EEG	Electroencephalogram
EIF	Exponential integrate-and-fire
EKG	Electrocardiogram
EPSP	Excitatory postsynaptic potential
FPGA	Field-Programmable Gate Array
GPU	Graphics processing unit
H-H	Hodgkin-Huxley
IF	Integrate-and-Fire
IoT	Internet of Things
IPSP	Inhibitory postsynaptic potential
IZH	Izhikevich
LIF	Leaky Integrate-and-Fire
LSTM	Long Short-term Memory
LTD	Long-Term Depression
LTP	Long-Term Potentiation
MB	Megabyte
MRI	Magnetska rezonancija
PRISMA	Preferred Reporting Items for Systematic Reviews and Meta-Analyses
ReLU	Rectified Linear Unit
RF-IZH	Resonate-and-Fire Izhikevich
R-STDP	Reward-Modulated Spike-Timing-Dependent Plasticity
SNN	Spiking Neural Network
SOTA	State of the art
STDP	Spike-Timing-Dependent Plasticity
STP	Short-Term Plasticity
TD-STDP	Temporal-Difference Spike-Timing-Dependent Plasticity
UI	Umjetna inteligencija
WTA	Winner-Take-All

Sažetak

Ovaj rad bavi se analizom razvoja modela neurona, od *klasičnih* umjetnih neurona do biološki inspiriranih modela i impulsnim neuronskim mrežama (engl. *Spiking Neural Networks*, SNN). U radu je prikazan povijesni razvoj neuronskih mreža, njihov matematički temelj te ključna ograničenja klasičnih pristupa, osobito u pogledu energetske učinkovitosti i izostanka vremenske dimenzije u obradi informacija.

Poseban fokus stavljen je na analizu triju dominantnih biofizičkih modela neurona: *Leaky Integrate-and-Fire* (LIF), Izhikevičev modela te biološki najprecizniji Hodgkin-Huxley (H-H) model. Obradene su osnovne karakteristike SNN-ova, uključujući složenosti neuronskog modela, načine kodiranja informacija, topologije mreža te metoda učenja poput plastičnosti ovisne o vremenu impulsa (STDP) i surogatnih gradijenata. Rad također daje pregled postojećih implementacijskih okvira i neuromorfni hardverskih rješenja, naglašavajući njihov značaj za razvoj energetske učinkovitih sustava.

Analiza literature ukazuje na izrazitu dominaciju LIF modela koji postiže vrhunske rezultate (SOTA), uz energetske potrošnje i do tri reda veličine manje u usporedbi s ANN modelima. Ipak, istraživanje identificira nedostatak standardiziranih metodologija za usporedbu različitih modela neurona i SNN sustava, kao i na ograničenja u evaluaciji koja često ne uzima u obzir njihove vremenske i energetske prednosti.

Zaključno, rad ističe da impulsne neuronske mreže predstavljaju obećavajući pristup za budući razvoj umjetne inteligencije, posebno u kontekstu sustava s ograničenim resursima kao što su *edge* uređaji i Internet stvari. Međutim, njihova šira primjena i učinkovitost uvelike ovisi o izboru modela neurona, načinu učenja i uvjetima evaluacije.