

EKONOMICKÁ UNIVERZITA V BRATISLAVE
FAKULTA HOSPODÁRSKEJ INFORMATIKY

Evidenčné číslo: 103004/I/2015/36066205684802564

**ŠTÚDIUM VZŤAHU MEDZI REKURENTNOU
NEURÓNOVOU SIEŤOU S LOGICKÝMI NEURÓNMI
A KONEČNOSTAVOVÝM ZARIADENÍM (AUTOMATOM)**

Diplomová práca

EKONOMICKÁ UNIVERZITA V BRATISLAVE
FAKULTA HOSPODÁRSKEJ INFORMATIKY

**ŠTÚDIUM VZŤAHU MEDZI REKURENTNOU
NEURÓNOVOU SIEŤOU S LOGICKÝMI NEURÓNMI
A KONEČNOSTAVOVÝM ZARIADENÍM (AUTOMATOM)**

Diplomová práca

Študijný program: Manažérske rozhodovanie a informačné technológie

Študijný odbor: 6258 Kvantitatívne metódy v ekonómii

Školiace pracovisko: Katedra aplikovanej informatiky

Vedúci záverečnej práce/školiteľ: RNDr. Eva Rakovská, PhD.

Bratislava 2015

Bc. Katarína Novotná

Čestne prehlasujem, že som túto diplomovú prácu vypracovala samostatne, pod odborným vedením RNDr. Eva Rakovská, PhD., s použitím uvedenej literatúry.

Dátum:

.....

Bc. Katarína Novotná

ABSTRAKT

NOVOTNÁ, Katarína: Štúdium vzťahu medzi rekurentnou neurónovou sieťou s logickými neurónmi a konečnostavovým zariadením (automatom) – Ekonomická univerzita v Bratislave. Fakulta hospodárskej informatiky – RNDr. Eva Rakovská, PhD. – Bratislava: FHI, 2015, 57 s.

Neurónové siete s logickými neurónmi môžu byť simulované konečnostavovým výpočtovým zariadením (automatom). Táto dôležitá vlastnosť platí aj v inverznom poradí, každé konečnostavové zariadenie môže byť simulované neurónovou sieťou s logickými neurónmi.

Predložená práca nadväzuje na bakalársku prácu, ktorá sa zaoberala skúmaním toho, že každá neurónová sieť je simulovaná konečnostavovým zariadením (automatom) tak, že obe zariadenia produkujú rovnaký výstupný reťazec znakov ako odozvu na vstupné znaky. Zároveň skúmala, že každé konečnostavové zariadenie (automat) je simulované neurónovou sieťou tak, že obe zariadenia produkovali rovnaký výstupný reťazec znakov ako odozvu na vstupné znaky.

V predloženej práci sa skúma konečnostavové zariadenie s viac ako dvoma stavmi simulované neurónovou sieťou.

ABSTRACT

NOVOTNÁ, Katarína: Study of the relationship between a recurrent neural network with logical neurons and a finite-state machine (automaton) – University of Economics in Bratislava. Faculty of economic informatics – RNDr. Eva Rakovská, PhD. – Bratislava: FHI, 2015, 57 p.

Neural networks with logical neurons can be simulated by finite-state machine (automaton). This important property is valid also vice versa; every finite-state machine can be simulated by a neural network with logical neurons.

The presented thesis builds on bachelor thesis, which investigates that every neural network can be simulated by a finite-state machine (automaton) by the assumption that both systems produce an identical output string of characters as a response to input characters.

The presented thesis examines finite-state machine with more than two states simulated by neural network.

Obsah

1	Úvod.....	1
1.1	Úvod do neurónových sietí.....	1
1.2	Použitie neurónových sietí v ekonómii	2
1.3	Cieľ práce	5
1.4	Aktuálna situácia a výsledky predošlého štúdia	7
2	Prevod konečnostavového automatu s maximálne dvoma stavmi na neurónovú sieť. 9	
2.1	Zadefinovanie konečnostavového automatu s dvoma stavmi	9
2.2	Tvorba neurónovej siete	12
2.3	Overenie ekvivalencie rekurentnej neurónovej siete s konečnostavovým automatom príkladom.....	14
3	Prevod konečnostavového automatu s N stavmi na neurónovú sieť	17
3.1	Zadefinovanie konečnostavového automatu s tromi stavmi	17
3.2	Tvorba neurónovej siete	20
3.2.1	Výstupná a prechodová vrstva pre každý stav konečnostavového automatu 20	
3.2.2	Prechodová vrstva pre každý stav konečnostavového automatu a samostatná výstupná funkcia	32
3.2.3	Prechodová vrstva pre každý stav konečnostavového automatu a ich použitie pre priamy výstup	37
3.3	Porovnanie rôznych spôsobov prevodu konečnostavového automatu na neurónovú sieť	42
4	Implementácia.....	47
4.1	Výber programovacieho jazyka.....	47
4.2	Zadávanie vstupných parametrov	47
4.3	Vykresľovanie rekurentnej neurónovej siete.....	49
5	Záver	53

Literatúra	55
Príloha A Prevod neurónovej siete na konečnosťový automat.....	58
A.1 Zadefinovanie neurónovej siete.....	58
A.2 Tvorba konečnosťového automatu	60
A.3 Zhodnotenie	69
Príloha B Inštalčná príručka	70
Príloha C Používateľská príručka.....	71
Príloha D Obsah disku CD	74

Zoznam obrázkov

Obr. 2.1: Konečnosťavový automat s výstupom	9
Obr. 2.2: Schematická reprezentácia konečnosťavového automatu reprezentovaného prechodovou a výstupnou funkciou	13
Obr. 2.3: Prechodová a výstupná funkcia reprezentovaná neurónovými sieťami	13
Obr. 2.4: Rekurentná neurónová sieť ekvivalentná s konečnosťavovým automatom.....	14
Obr. 2.5: Rekurentná neurónová sieť	15
Obr. 3.1: Konečnosťavový automat s troma stavmi	17
Obr. 3.2: Neurón reprezentujúci prechodovú funkciu pri vstupnej hodnote 0.....	21
Obr. 3.3: Neurón reprezentujúci prechodovú funkciu pri vstupnej hodnote 1.....	22
Obr. 3.4: Výstupný neurón pre stav S_0	23
Obr. 3.5: Výstupný neurón pre stav S_1	24
Obr. 3.6: Výstupný neurón pre stav S_2	26
Obr. 3.7: Výstupný neurón rekurentnej neurónovej siete	27
Obr. 3.8: Rekurentná neurónová sieť ekvivalentná so zadaným konečnosťavovým automatom.....	28
Obr.: 3.9: Rekurentná neurónová sieť s popismi	29
Obr. 3.10: Výstupný neurón pre štvrtý typ stavu	31
Obr. 3.11: Rekurentná neurónová sieť ekvivalentná so zadaným konečnosťavovým automatom.....	34
Obr. 3.12: Rekurentná neurónová sieť s popismi.....	35
Obr. 3.13: Rekurentná neurónová sieť ekvivalentná so zadaným konečnosťavovým automatom.....	39
Obr. 3.14: Rekurentná neurónová sieť s popismi.....	40
Obr. 3.15: Neurónová sieť s nulovým výstupom v každom prechode.....	43
Obr. 4.1: Rekurentná neurónová sieť vygenerovaná programom.....	52
Obr. A.1: Rekurentná neurónová sieť	59

Obr. A.2: Prvý prechod konečnosťavového automatu	62
Obr. A.3: Tretí prechod konečnosťavového automatu	63
Obr. A.4: Konečnosťavový automat reprezentujúci neurónovú sieť	64
Obr. A.5: Konečnosťavový automat s prechodmi zo stavu S100 pri vstupe 0 s rôznymi výstupmi v čase t-1	66
Obr. A.6: Konečnosťavový automat reprezentujúci neurónovú sieť pri akýchkoľvek vstupoch	67
Obr. C.1: Úvodná stránka programu	71
Obr. C.2: Formulár pre zadanie počtu stavov automatu	71
Obr. C.3: Formulár pre zadanie prechodovej a výstupnej funkcie	72
Obr. C.4: Príklad výstupu programu	72
Obr. C.5: Chybové oznámenia vo formulári	73

Zoznam tabuliek

Tabuľka 2.1: Prechodová a výstupná funkcia konečnosťového automatu	10
Tabuľka 2.2: Prechodová funkcia konečnosťového automatu.....	10
Tabuľka 2.3: Výstupná funkcia konečnosťového automatu	11
Tabuľka 2.4: Karnaughova mapa pre prechodovú funkciu.....	11
Tabuľka 2.5: Karnaughova mapa pre výstupnú funkciu.....	12
Tabuľka 2.6: Výstup a prechod konečnosťového automatu na základe vstupných dát. 14	
Tabuľka 2.7: Stavby neurónovej v čase rekurentnej neurónovej siete.....	16
Tabuľka 3.1: Prechodová a výstupná funkcia konečnosťového automatu	18
Tabuľka 3.2: Prechodová funkcia konečnosťového automatu.....	19
Tabuľka 3.3: Výstupná funkcia konečnosťového automatu	19
Tabuľka 3.4: Karnaughova mapa pre prechodový neurón pri vstupnej hodnote 0.....	21
Tabuľka 3.5: Karnaughova mapa pre prechodový neurón pri vstupnej hodnote 1	22
Tabuľka 3.6: Karnaughova mapa pre výstupný neurón stavu S_0	23
Tabuľka 3.7: Karnaughova mapa pre výstupný neurón stavu S_1	24
Tabuľka 3.8: Karnaughova mapa pre výstupný neurón stavu S_2	25
Tabuľka 3.9: Upravená Karnaughova mapa pre výstupný neurón stavu S_2	25
Tabuľka 3.10: Stavby neurónov v čase	30
Tabuľka 3.11: Karnaughova mapa pre výstupnú funkciu.....	33
Tabuľka 3.12: Upravená Karnaughova mapa pre výstupnú funkciu	33
Tabuľka 3.13: Stavby neurónov v čase	36
Tabuľka 3.14: Stavby neurónov v čase	41
Tabuľka A.1: Aktivita neurónov v čase – vstup a výstup siete.....	59
Tabuľka A.2: Aktivita neurónov v čase – stavby skrytých neurónov.....	61
Tabuľka A.3: Aktivita neurónov v čase $t=1$ a $t=2$ – prechod konečnosťového automatu	61

Tabuľka A.4: Aktivita neurónov v čase $t=3$ a $t=4$	62
Tabuľka A.5: Zisťovanie prechodu zo stavu $S000$ pri vstupe 0 a výstupe 0.....	65
Tabuľka A.6: Zisťovanie prechodu zo stavu $S000$ pri vstupe 0 a výstupe 1.....	65
Tabuľka A.7: Zisťovanie prechodu zo stavu $S100$ pri vstupe 0 a výstupe 0.....	65
Tabuľka A.8: Zisťovanie prechodu zo stavu $S100$ pri vstupe 0 a výstupe 1.....	65
Tabuľka A.9: Krok 1 – overenie nedosiahnuteľného stavu $S001$	68
Tabuľka A.10: Krok 2 – overenie nedosiahnuteľného stavu $S001$	69

1 Úvod

1.1 Úvod do neurónových sietí

Už od nepamäti fascinovalo ľudí predpovedať veličiny, ktoré na prvý pohľad pôsobia chaoticky. S využitím poznatkov z matematickej a štatistickej analýzy, lineárnej algebry spolu so správnou dávkou ľudského úsilia sa zákonitosti, medzi javmi, pôsobiacimi chaoticky dajú korektne matematicky vyjadriť a použiť ako základ pre predpovedanie stavov v budúcnosti.

Lineárne modely sa stali pokročilou doménou rozhodovania investorov po celé dekády, hlavne pre svoju relatívne jednoduchú interpretáciu skúmaných javov. Postupom času však lineárne modely nepostačovali požiadavkám investorov a začala sa éra vývoja modelov, schopných zachytiť nelineárne vzťahy skúmaných systémov.

Neurónové siete vplyvom rýchleho rozvoja výpočtovej techniky sa využívajú čoraz viac v experimentálnych úlohách ako aj v praxi. Aplikačné úlohy typu spracovania informácií, klasifikácie vzorov alebo situácií, optimalizačné problémy, predikčné úlohy nachádzajú uplatnenie v rôznych oblastiach priemyslu, managementu, finančníctva, telekomunikácie, vojenskej techniky, zdravotníctva a inde.

Simulácia stavov reálneho sveta pomocou neurónových sietí patrí medzi najsofistikovanejšie metódy simulácie stavov reálneho sveta (Kvasnička, 1997). Neurónové siete odhaľujú dynamické nelineárne procesy, odohrávajúce sa na burze v reálnom čase a v určitých prípadoch zohľadňujú taktiež časopriestorovú štruktúru takýchto javov. Implementácia fragmentu umelej inteligencie pri predpovedi vývoja podkladového aktíva je podmienený výberom vhodnej štruktúry použitej siete. Za posledné roky bolo navrhnutých viacero modelov neurónových sietí, ktoré boli optimalizované na riešenie konkrétneho problému (Sinčák, 1996). Najviac sa však v praxi uplatnili modely rekurentných a dopredných neurónových sietí.

Zo štruktúry vlastností dopredných sietí vyplýva, že nie sú schopné riešiť úlohy s časovou súvislosťou, nakoľko neobsahujú možnosť pamätať si predchádzajúce stavy.

Problémy spôsobené absenciou spracovávania informácií z diskretných časových úsekov v minulosti podnietili vývoj nových typov sietí - rekurentných neurónových sietí (Recurrent Neural Networks, RNN). Rekurentná neurónová sieť predstavuje dynamicky zložitejší systém doprednej neurónovej siete (FFNN). Signál v FFNN je šírený iba jedným smerom – dopredu, sieť nie je schopná pracovať s históriou, nevie rozoznať v dátach časovú závislosť. Aby FFNN dokázala zachytiť túto časovú závislosť, je potrebné ju mierne upraviť (Sinčák, 1996). RNN je neurónová sieť, ktorej podmnožina neurónov (tzv. rekurentných neurónov) uchováva informáciu o svojich aktivitách z predošlých stavov v kontextovej vrstve (Mandic, 2001). O výstupe z neurónu nerozhoduje len momentálny vstup v čase t ale aj vstupy z minulosti. Učiace algoritmy musia brať do úvahy túto skutočnosť. Rekurentné siete sú ďalej schopné uchovávať stavové informácie časového kontextu a ich použitie je vhodné najmä v úlohách, kde je nevyhnutné zohľadniť aj časovú štruktúru v dátach. Pri použití tohto typu neurónovej siete je najvhodnejšia aplikácia v úlohách, kde sa kladie dôraz na časovú zložku ukrytú v dátach.

Uplatnenie rekurentných neurónových sietí môžeme rozdeliť do nasledovných kategórií (Koščak, 2009):

- *klasifikačné – vhodné na riešenie asociačných úloh s časovou závislosťou - sieť musí rozhodnúť, či daná postupnosť vstupov patrí do nejakej triedy,*
- *predikčné úlohy – v danej štruktúre časovo závislých dát má neurónová sieť nájsť závislosť medzi dátami a na základe časového úseku určiť hodnotu v nasledujúcom čase,*
- *generatívne úlohy – zložitejšia verzia predikcie, kde sieť na základe pozorovania úseku dát musí vedieť pokračovať v časovom rade dát vzhľadom na základnú tendenciu dát skrytú v danej postupnosti.*

1.2 Použitie neurónových sietí v ekonómii

Neurónové siete sú alternatívou k použitiu diskriminačnej analýzy vo finančnej analýze. Nevyžaduje hlboké matematicko-štatistické vedomosti ani existenciu žiadnych predpokladov. Viacrozmerná diskriminačná analýza naopak predpokladá napr. normálne rozdelenie hodnôt ukazovateľov, nezávislosť ukazovateľov.

Už od čias Beavera (1966) a Altmana (1968) bol značný záujem o využitie pomerových finančných ukazovateľov pre potreby finančnej analýzy ex-ante. Avšak, tieto predikčné metódy diskriminačnej analýzy boli kritizované najmä pre ich linearitu a obmedzujúce predpoklady (Neves, Vieira, 2006, s. 254). Tieto nedostatky sú odstránené práve u neurónových sietí, ktoré boli pôvodne navrhnuté preto, aby vedcom a výskumníkom pochopili fungovanie ľudského mozgu. V súčasnom ekonomickom prostredí môžeme vidieť, že neurónové siete sú postupne integrované do riešenia problémov z rôznych oblastí a stávajú sa stále dôležitejším analytickým nástrojom.

Aplikácia neurónových sietí pre klasifikáciu podnikov do skupiny prosperujúcich, resp. neprosperujúcich podnikov je pomerne nová, ich využitie na tento účel je predmetom výskumu iba od začiatku deväťdesiatych rokov minulého storočia (Lesáková a kol., 2007, s. 82). Za priekopníkov v oblasti využívania neurónových sietí pre potreby finančnej predikcie sú považovaní Odom a Sharda, ktorí v roku 1990 aplikovali metódu „backpropagation“ na predikovanie bankrotu podniku, pričom ich model dosiahol lepšie výsledky ako metódy porovnávanej diskriminačnej analýzy, ktoré boli dovtedy používané na riešenie danej problematiky (Shah, Murtaza, 2000, s. 80). Odom a Sharda použili ako vstupy do neurónových sietí pomerové finančné ukazovatele z výpočtu Altmanovho Z-skóre. Autori spoľahlivosť neurónových sietí porovnávali so spoľahlivosťou viacrozmernej diskriminačnej analýzy. Výskum bol uskutočnený na vzorke 128 solventných i bankrotujúcich anglických firmách.

Presnosť neurónových sietí z hľadiska správneho zaradenia:

- *bankrotujúcich podnikov bola 77,8 % - 81,5 %,*
- *solventných podnikov bola 78,6 % - 85,7 %.*

Presnosť viacrozmernej diskriminačnej analýzy bola nasledovná:

- *bankrotujúce podniky 59,3 % - 70,4 %,*
- *solventné podniky 78,6 % - 85,7 %. (Atiya, 2001, s. 930).*

Na základe vyššie uvedených výsledkov sa v danom výskume neurónové siete v porovnaní s viacrozmernou diskriminačnou analýzou ukázali ako analytický nástroj s

lepšou predikčnou schopnosťou, čo následne vyvolalo zvýšený záujem o ich využívanie na účel predikovania úpadku podniku.

Predikovanie úpadku podniku ale nie je jediné praktické využitie neurónovej siete v ekonómii. Ústav informatiky Ekonomickej fakulty na Mendelovej univerzite v Brne, vydal celú radu publikácií na tému využitia neurónových sietí v ekonomických oblastiach. Ich výskum v tejto oblasti na ÚI PEF MZLU v Brne sleduje prevažne ekonomickú aplikáciu s cieľom podpory podnikových rozhodovacích procesov na strategickej úrovni riadenia.

Témy, ktorým sa venovali na ÚI PEF MZLU v danej oblasti:

- *Predikcia vývoja ekonomických ukazovateľov - Predikcia budúceho vývoja ekonomických ukazovateľov je riešená v rámci modelu pre odhad trendov viacvrstvovou perceptrónovou neurónovou sieťou (MLP) a využitím evolučných algoritmov ako obecných optimalizačných metód. Novinkou je riešenie regresnej úlohy pomocou gramatickej evolúcie.*
- *Implementácia podnikových rozhodovacích procesov a pravidiel - Podpora rozhodovacích procesov na úrovni podnikových pravidiel je riešená expertnými systémami a fuzzy expertnými systémami, najčastejšie s využitím prostredia systému LPA WinProlog.*
- *Diagnostický systém vyhodnotenia podnikovej finančnej situácie - Úlohy hodnotenia finančnej situácie podniku sú prevažne klasifikačným typom úlohy. Cieľom je zostavenie systému umožňujúceho identifikovať finančnú situáciu podniku (hlavne negatívnu situáciu) rovnako ako finančný analytik. Pozornosť je zameraná na sektor malých a stredných podnikov. Využívané sú viacvrstvové perceptrónové neurónové siete naučené na základe bázy vzorkov zostavenej expertom, snahou je získať aj bázu s reálnymi ekonomickými výsledkami. V prípade definovaných pravidiel sú využívané taktiež expertné a fuzzy expertné systémy.*
- *Identifikácia preferencie zákazníkov - Výskumný smer zameraný na oblasti marketingu a riadenia vzťahov so zákazníkmi v spojení s problematikou poradenských systémov a sémantického vyhľadávani v dokumentoch. Jednou z výskumných aktivít je simulácia zákazníkovej úvahy pri nákupe prostredníctvom technológií neurónových sietí.*

- *Aplikácia podnikovej inteligencie - Implementácia technológií umelé inteligencie v rozhodovacom procese súvisí a v mnohých smeroch nadväzuje na aplikáciu podnikovej inteligencie z oblasti integrácie business intelligence, dátových skladov, OLAP nástrojov a metód data miningu.*

Veľmi zaujímavou je aj práca „Use of Artificial Neural Networks to Predict The Business Success or Failure of Start-Up Firms“ (po anglicky vo voľnom preklade: Použitie neurónových sietí na predikovanie úspechu alebo neúspechu start-up spoločností), ktorú vypracoval Dr. Francisco Garcia Fernandez. Momentálny trend v oblasti podnikania na posledné roky sú start-upy. Bohužiaľ, na ich začiatku nikto nevie s určitosťou povedať, či bude start-up úspešný alebo nie. Táto skutočnosť robí vrásky na čele nie len samotným start-upistom alebo aj potencionálnym investorom. Predikcia úspechu či neúspechu start-upu ešte pred jeho samotným spustením by ušetrila úsilie, financie a čas zúčastneným osobám a pomohla by tieto zdroje využiť na start-up s lepšou šancou na úspech. Doktor v závere svojej práce píše: „...it is expected to have a very useful tool to predict the business success or failure of start-up firms“ [12]

Ďalšie možnosti uplatnenia neurónových sietí v business otázkach popísali Kate A. Smith a Jatinder N.D. Gupta v ich publikácii Neural networks in business. V práci popísali využitie neurónových sietí v oblasti marketingu, nákupu, predaja, bankového sektora, finančného sektora, poistenia, telekomunikácií, data minningu a mnoho ďalších.

1.3 Cieľ práce

Z kapitoly 1.2 vyplýva, že neurónové siete sú dnes už neodmysliteľnou súčasťou ekonómie a ekonomických procesov. Neustály vývoj informačných technológií posúva hranice použiteľnosti neurónových sietí na riešenie ekonomických problémov. Ľudský faktor sa postupne stáva jedinou prekážkou pre plné využívanie neurónových sietí. Práca s neurónovými sieťami je pre väčšinu ľudí totiž zložitá. Bohužiaľ neexistuje vedecká štúdia, ktorá by sa zaoberala rozdielom náročnosti výučby neurónových sietí v porovnaní s výučbou konečnostavových automatov. Na webovej stránke quora.com sa zato nachádza zaujímavá odpoveď na otázku „How would you explain Finite State Machines to a layman?“ (po anglicky vo voľnom preklade: Ako by ste vysvetlili konečnostavový automat

laikovi?). V rozsiahlej odpovedi Eric P. okrem vysvetlenia konečnosťavového automatu pridal aj odsek, ktorý stojí za zamyslenie: „Why study them? They are a simple and obvious thing to do, but I've discovered that there are a lot of developers in the field who have no idea how to use them.“ (po anglicky vo voľnom preklade: Prečo to študovať? Automat je jednoduchý a jasný, ale zistil som, že je tu mnoho programátorov v obore, ktorý nemajú tušenia ako ho použiť.)

Ako je všeobecne známe, konečnosťavové automaty sú na pochopenie, na implementáciu aj na výučbu jednoduchšie oproti neurónovým sieťam. No taktiež si málokto uvedomuje ich potenciál. Pokiaľ ale hovoríme o neurónových sieťach, ich využitie je dnes už známe širokej verejnosti a ich popularita stále stúpa. Cieľom tejto diplomovej práce je preto dokázať ekvivalenciu konečnosťavového automatu a neurónovej siete, čo by malo za následok uľahčenie práce ekonomickým pracovníkom s neurónovými sieťami, zjednodušiť a urýchliť vývoj neurónových sietí zameraných na riešenie ekonomických otázok. V neposlednom rade je cieľom tejto práce aj prispieť do vývoja a výučby kognitívnej vedy na Slovensku ako aj v zahraničí.

Na zlepšenie efektivity ekonomických pracovníkov by preto mohol prispieť práve softvér, ktorý prevádza konečnosťavové automaty na neurónové siete. Následne pracovníci, ktorý sa podieľajú na vývoji ekonomických softvérov s použitím neurónových sietí alebo pracujúci s týmto softvérom, by už neboli nútení pracovať a rozmýšľať na úrovni neurónových sietí, ale mohli by sa posunúť na jednoduchšie a ľahšie pochopiteľné konečnosťavové automaty a urýchliť tým pracovné postupy a znížiť chybovosť.

Návrh softvéru, ktorý je cieľom tejto diplomovej práce, by prispel k rýchlejšiemu vývoju ekonomických softvérov založených na neurónových sieťach a zároveň by uľahčil pochopenie procesov ekonomickými pracovníkmi.

Teoretický a praktický návod matematického aj programátorského prevodu konečnosťavového automatu s viac ako 2 stavmi na neurónovú sieť taktiež výrazne prispeje k skvalitneniu výučby kognitívnej vedy nielen na Slovensku, nakoľko neexistuje vedecká práca zaoberajúca sa praktickou implementáciou takéhoto prevodu.

1.4 Aktuálna situácia a výsledky predošlého štúdia

Pokiaľ máme v úmysle implementovať ekvivalenciu, resp. praktický prevod, medzi konečnosťavým automatom a neurónovou sieťou, v prvom rade musíme zvoliť správny typ neurónovej siete pre tieto potreby. Ako už bolo naznačené v kapitole 1.1, rekurentné neurónové siete sú schopné reagovať nie len na základe vstupných dát ale aj na základe výsledkov v minulosti. Nielenže je táto schopnosť využívaná v mnohých ekonomických procesoch, ale zároveň nám umožňuje priblížiť sa ku konečnosťavému automatu, v ktorom sa po istých krokoch výpočet dostane do určitého stavu, ktorý postupuje ďalej nie len na základe ďalšieho vstupného parametru ale aj na základe aktuálnej pozície, na ktorú sa automat dostal určitou postupnosťou predošlých vstupných parametrov.

Dôkazu ekvivalencie konečnosťavého automatu a rekurentnej neurónovej siete a naopak sa venovalo už mnoho odborníkov [1], [15], [16], [17], [18], [19]. Dospeli k matematickým dôkazom ekvivalencie medzi rekurentnou neurónovou sieťou a konečnosťavým automatom a naopak. Praktickým prevodom sa bohužiaľ nikto z nich nevenoval. Dôvodom môže byť aj skutočnosť, že v dobe kedy sa tieto práce písali (1984-1992) výpočtová technika nebola na dostatočnej úrovni, aby plne zvládla praktický prevod konečnosťavého automatu na rekurentnú neurónovú sieť a naopak.

Bc. Katarína Novotná (autor tejto práce) počas bakalárskeho štúdia s pánom prof. Ing. Vladimírom Kvasničkom, DrSc. vypracovala bakalársku prácu na tému: Štúdium vzťahu medzi rekurentnou neurónovou sieťou s logickými neurónmi a konečnosťavým zariadením (automatom). Výsledkom vyššie uvedenej bakalárskej práce bol prevod konečnosťavého automatu s maximálne dvoma stavmi, ktorý je znázornený v kapitole 2 a prevod rekurentnej neurónovej siete na konečnosťavý automat, ktorý je popísaný v prílohe A. Pri vypracovávaní bakalárskej práce ale narazili na problém pri určovaní prechodových a výstupných funkcií konečnosťavého automatu, pokiaľ mal automat viac ako dva stavy. Pri konečnosťavom automate s viac ako dvoma stavmi už Karnaughova mapa nepostačovala na určenie prechodovej a výstupnej funkcie. Z tohto dôvodu je potrebné nájsť lepšiu cestu za prechodovou a výstupnou funkciou, čo je aj hlavnou témou tejto diplomovej práce. Z dôvodu veľkosti tejto problematiky autor bakalárskej práce nepokračoval ďalej v skúmaní prevodu konečnosťavého automatu na neurónovú sieť.

Cieľom diplomovej práce je nadviazať na vyššie spomenutú bakalársku prácu, navrhnúť a implementovať prevod konečnostavového automatu s viac ako dvoma stavmi na neurónovú sieť, čím sa zavŕši praktický prevod, resp. ekvivalencia, medzi neurónovými sieťami a konečnostavovým zariadením.

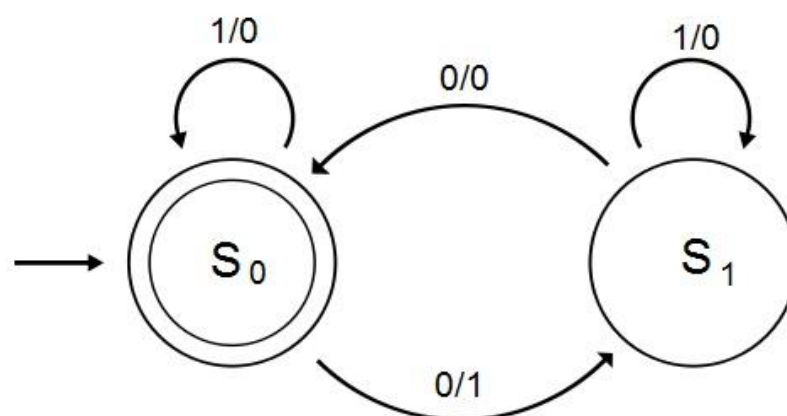
2 Prevod konečnostavového automatu s maximálne dvoma stavmi na neurónovú sieť

Teóriu prevodu konečnostavového automatu na rekurentnú neurónovú sieť zadefinoval V. Kvasnička vo svojej knihe, kde tvrdí že: „Každý konečnostavový stroj s výstupom (Mealyho automat) môže byť reprezentovaný ekvivalentnou rekurentnou neurónovou sieťou“ [1].

Prevod bol uskutočnení extrakciou prechodovej a výstupnej funkcie z konečnostavového automatu (kapitola 2.1), následnou reprezentáciou týchto funkcií pomocou logických neurónov a ich substitúciou do rekurentnej neurónovej siete (kapitola 2.2).

2.1 Zadefinovanie konečnostavového automatu s dvoma stavmi

V prvom kroku vytvoríme tabuľku (tabuľka 2.1) prechodovej a výstupnej funkcie konečnostavového automatu na obrázku 2.1.



Obr. 2.1: Konečnostavový automat s výstupom

	Prechodová funkcia		Výstupná funkcia	
	0	1	0	1
S_0	S_1	S_0	1	0
S_1	S_0	S_1	0	0

Tabuľka 2.1: Prechodová a výstupná funkcia konečnosťového automatu

Zistené hodnoty si rozpíšeme do tabuľky pre zistenie binárnych reprezentácií funkcie. V prechodovej funkcii rozpisanej v tabuľke 2.2 zameníme výstupné stavy za binárne hodnoty, ktoré reprezentujú výstup prechodovej funkcie. Výstupná funkcia je rozpísaná v tabuľke 2.3.

Východiskový stav	Vstup	Binárna reprezentácia	Výstupný stav	Binárna reprezentácia výstupu
S_0	0	0,0	S_1	1
S_0	1	0,1	S_0	0
S_1	0	1,0	S_0	0
S_1	1	1,1	S_1	1

Tabuľka 2.2: Prechodová funkcia konečnosťového automatu

Východiskový stav	Vstup	Binárna reprezentácia	Výstup
S_0	0	0,0	1
S_0	1	0,1	0
S_1	0	1,0	0
S_1	1	1,1	0

Tabuľka 2.3: Výstupná funkcia konečnostavového automatu

Následne určíme funkciu, ktorú reprezentuje prechodová a výstupná funkcia konečnostavového automatu. Môžeme pri tom použiť Karnaughovu mapu. Do Karnaughovej mapy zapíšeme hodnoty z tabuľky funkcie tak, že prvá hodnota binárnej reprezentácie funkcie predstavuje 1.stĺpec (X) v Karnaughovej mape a druhá hodnota binárnej reprezentácie funkcie predstavuje 1. riadok (Y) v mape. Binárnu reprezentáciu výstupu prechodovej funkcie, či výstup funkcie pri výstupnej funkcii konečnostavového automatu, zapíšeme do obsahu tabuľky, respektíve ako hodnoty Karnaughovej mapy.

X\Y	0	1
0	1	0
1	0	1

Tabuľka 2.4: Karnaughova mapa pre prechodovú funkciu

Z tabuľky 2.4, Karnaughova mapa pre prechodovú funkciu, vyplýva, že výstup 1 nastáva iba v prípade, ak X a zároveň Y sú 0 alebo X a zároveň Y sú 1. Z tejto skutočnosti

môžeme vyhodnotiť tvar prechodovej funkcie konečnosťového automatu z obrázka 2.1. Prechodová funkcia konečnosťového automatu má tvar:

$$(NOT X AND NOT Y) OR (X AND Y)$$

Rovnakým spôsobom vyhodnotíme aj výstupnú funkciu konečnosťového automatu.

X\Y	0	1
0	1	0
1	0	0

Tabuľka 2.5: Karnaughova mapa pre výstupnú funkciu

Z tabuľky 2.5 vyplýva, že výstup 1 nastáva iba v prípade ak X a zároveň Y sú 0. Výstupná funkcia konečnosťového automatu má tvar:

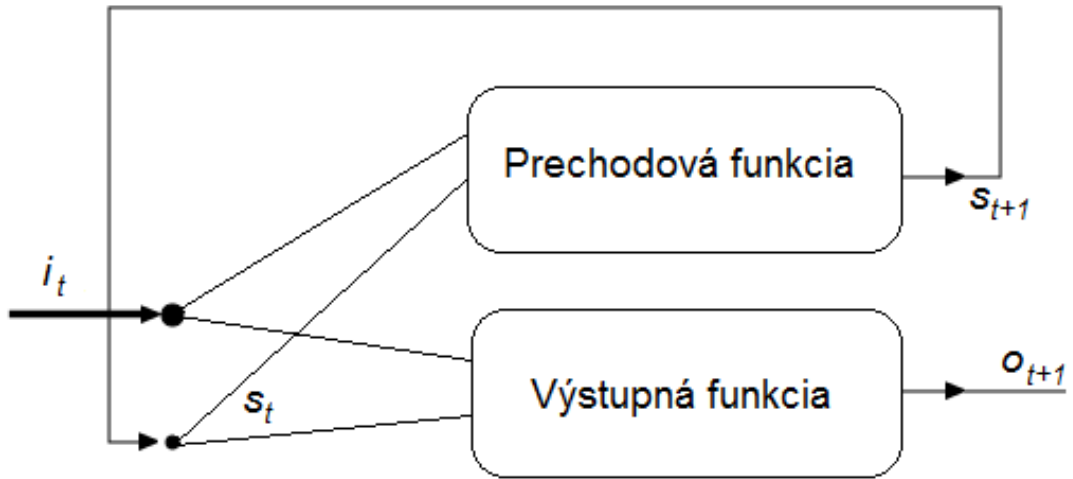
$$NOT X AND NOT Y$$

2.2 Tvorba neurónovej siete

Konečnosťový automat má 1 pásku vstupných hodnôt. Na implementáciu prechodovej a výstupnej funkcie potrebujeme ale dve hodnoty X a Y. Tieto znaky si reprezentujeme v obrázku 2.2 ako „i“ – vstupná hodnota a „s“ – výstupné hodnoty prechodovej funkcie. Výstup neurónovej siete, ktorý je zároveň aj výstupom konečnosťového automatu definujeme ako „o“. Tieto znaky sú medzi sebou v dvoch funkčných reláciách, kde f je prechodová funkcia a g je výstupná funkcia konečnosťového automatu:

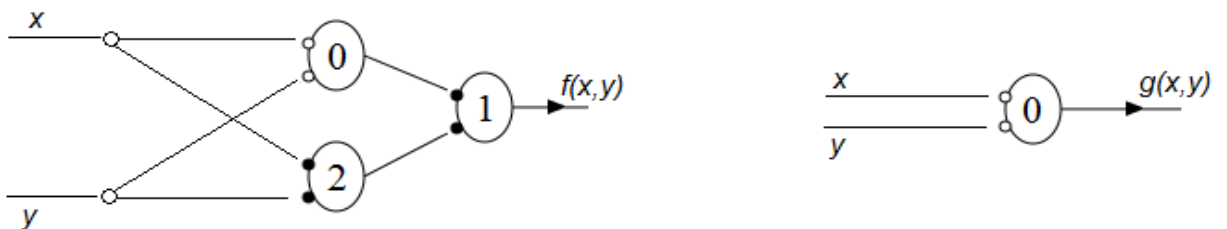
$$s^{(t+1)} = f (s^{(t)}, i^{(t)})$$

$$o^{(t+1)} = g (s^{(t)}, i^{(t)})$$



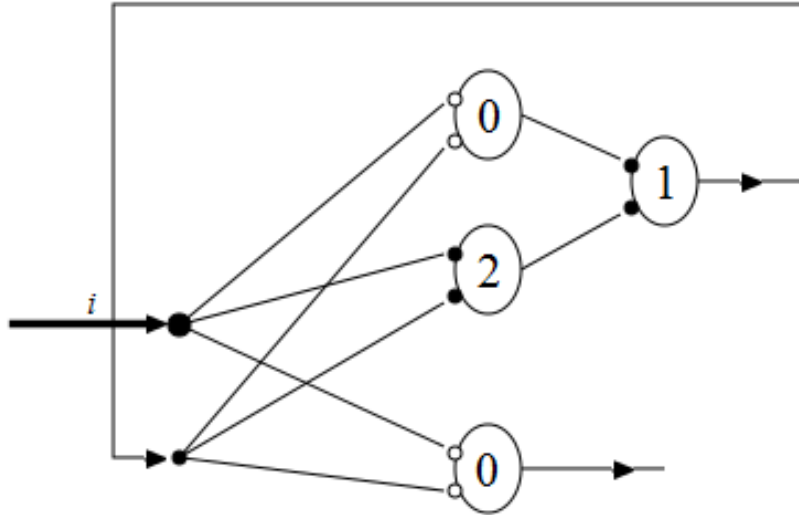
Obr. 2.2: Schematická reprezentácia konečnosťového automatu reprezentovaného prechodovou a výstupnou funkciou

Prechodovú a výstupnú funkciu konečnosťového automatu na obrázku 2.1 špecifikujeme neurónovou sieťou (obrázok 2.3).



Obr. 2.3: Prechodová a výstupná funkcia reprezentovaná neurónovými sieťami

Substitúciou týchto neurónových sietí z obrázku 2.3 do schémy na obrázku 2.2 získame rekurentnú neurónovú sieť, ktorá je ekvivalentná s konečnosťovým automatom na obrázku 2.1.



Obr. 2.4: Rekurentná neurónová sieť ekvivalentná s konečnosťavým automatom

2.3 Overenie ekvivalencie rekurentnej neurónovej siete s konečnosťavým automatom príkladom

Vytvorenú rekurentnú neurónovú sieť porovnáme so zadaným konečnosťavým automatom pomocou príkladu. Neurónová sieť musí generovať rovnaké výstupné dáta ako konečnosťavý automat pri rovnakej vstupnej postupnosti. Vstupnou testovacou postupnosťou je postupnosť (110110101010).

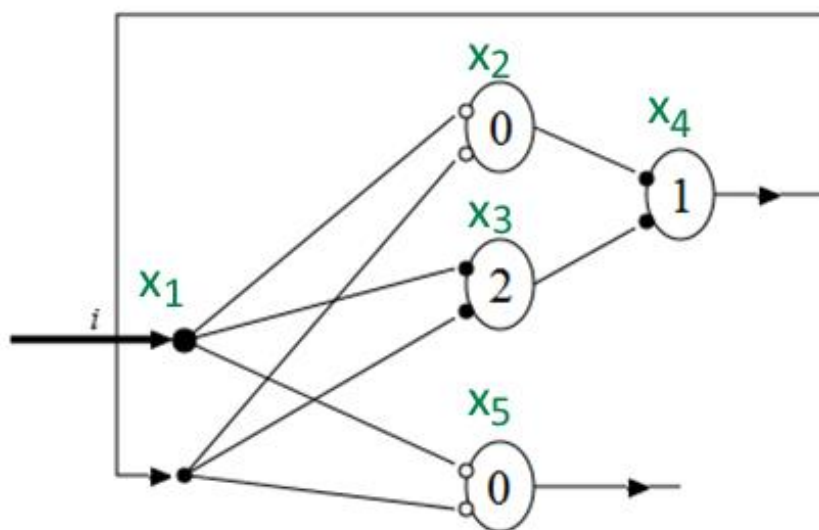
V prvej kroku určíme prechodné a výstupné hodnoty konečnosťavého automatu (tabuľka 2.6).

Vstup	1	1	0	1	1	0	1	0	1	0	1	0
Prechod	S_0	S_0	S_1	S_1	S_1	S_0	S_0	S_1	S_1	S_0	S_0	S_1
Výstup	0	0	1	0	0	0	0	1	0	0	0	1

Tabuľka 2.6: Výstup a prechod konečnosťavého automatu na základe vstupných dát

Zadefinujeme si v rekurentnej neurónovej sieti ekvivalentnej s konečnosťavým automatom jednotlivé neuróny. Neuróny v sieti sme označili x_1 až x_5 (obrázok 2.5). Vstup

výsledku prechodovej funkcie späť do neurónovej sieti nemusíte označovať samostatne, nakoľko vstup prechodovej funkcie v čase $t+1$ sa rovná stavu neurónu x_4 v čase t .



Obr. 2.5: Rekurentná neurónová sieť

Začiatočným stavom v konečnostavovom automate je stav S_0 , preto stanovíme počiatočný stav neurónu x_4 v čase $t-1$ nula. Následne stavy neurónov v čase pri vstupnej postupnosti (1101101010) zodpovedajú stavom v tabuľke 2.7.

x_1	x_2	x_3	x_4	x_5
1	0	0	0	0
1	0	0	0	0
0	1	0	1	1
1	0	1	1	0
1	0	1	1	0

0	0	0	0	0
1	0	0	0	0
0	1	0	1	1
1	0	1	1	0
0	0	0	0	0
1	0	0	0	0
0	1	0	1	1

Tabuľka 2.7: Stavy neurónovej v čase rekurentnej neurónovej siete

Z tabuľky 2.7 ako aj z obrázka 2.5 jasne vyplýva, že výstupná postupnosť sa nachádza v stĺpci x_5 a prechodová funkcia je reprezentovaná binárnymi hodnotami v stĺpci x_4 . Z tabuliek 2.6 a 2.7 vyplýva, že výstupná postupnosť je (001000010001) a prechodová funkcia je reprezentovaná binárnou postupnosťou (001110011001).

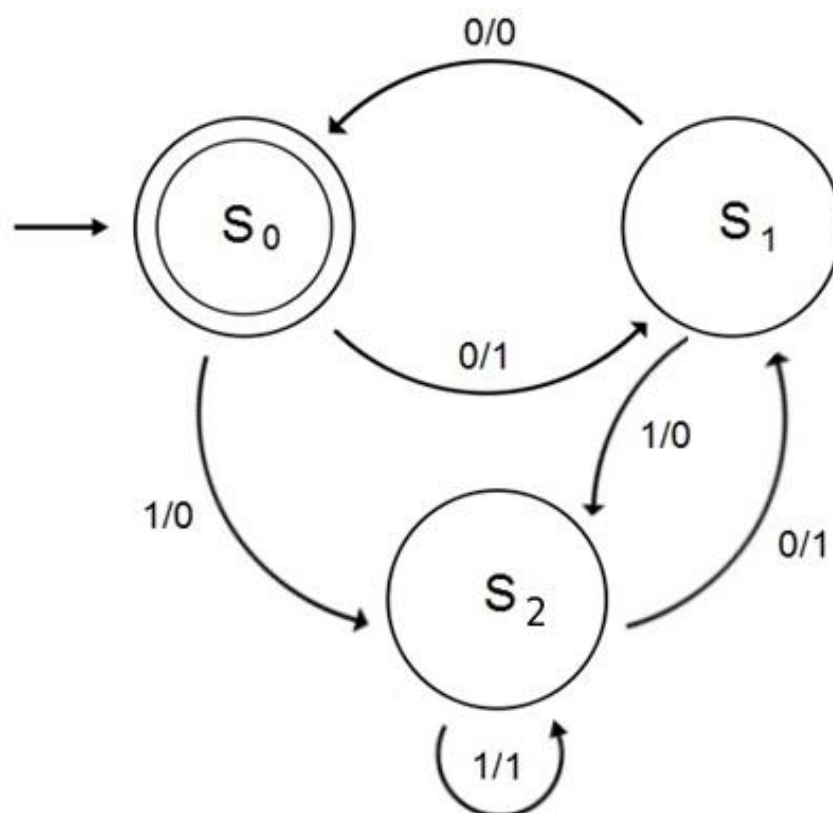
Výstup konečnostavového automatu je zhodný s výstupom rekurentnej neurónovej siete na základe čoho môžeme konštatovať, že rekurentná neurónová sieť na obrázku 2.5 je ekvivalentná s konečnostavovým automatom na obrázku 2.1.

3 Prevod konečnostavového automatu s N stavmi na neurónovú sieť

Prevod konečnostavového automatu na neurónovú sieť sme rozdelili na dva typy. Prvým typom je prevod konečnostavového automatu s maximálne dvoma stavmi (kapitola 2) a druhým typom je prevod konečnostavového automatu s viac ako dvoma stavmi. Dôvodom tohto rozdelenia je prechodová funkcia, čo je podrobnejšie vysvetlené v tejto kapitole priamo na príklade.

3.1 Zadefinovanie konečnostavového automatu s tromi stavmi

V prvom kroku vytvoríme tabuľku (tabuľka 3.1) prechodovej a výstupnej funkcie konečnostavového automatu s tromi stavmi na obrázku 3.1.



Obr. 3.1: Konečnostavový automat s troma stavmi

	Prechodová funkcia		Výstupná funkcia	
	0	1	0	1
S_0	S_1	S_2	1	0
S_1	S_0	S_2	0	0
S_2	S_1	S_2	1	1

Tabuľka 3.1: Prechodová a výstupná funkcia konečnostavového automatu

Zistené hodnoty si rozpíšeme do tabuľky pre zistenie binárnych reprezentácií funkcie. V prechodovej funkcii rozpisanej v tabuľke 3.2 zameníme výstupné stavy za binárne hodnoty, ktoré reprezentujú výstup prechodovej funkcie. Výstupná funkcia je rozpísaná v tabuľke 3.3.

Východiskový stav	Vstup	Reprezentácia	Výstupný stav	Reprezentácia výstupu
S_0	0	0,0	S_1	1
S_0	1	0,1	S_2	2
S_1	0	1,0	S_0	0
S_1	1	1,1	S_2	2

S_2	0	2,0	S_1	1
S_2	1	2,1	S_2	2

Tabuľka 3.2: Prechodová funkcia konečnostavového automatu

Východiskový stav	Vstup	Reprezentácia	Výstup
S_0	0	0,0	1
S_0	1	0,1	0
S_1	0	1,0	0
S_1	1	1,1	0
S_2	0	2,0	1
S_2	1	2,1	1

Tabuľka 3.3: Výstupná funkcia konečnostavového automatu

Z tabuľky 3.2 vyplýva, že nemôžeme prechodovú funkciu reprezentovať binárnymi hodnotami. Binárna reprezentácia nie je dostatočná pre prechodovú funkciu konečnostavového automatu s viac ako dvoma stavmi a toto je základným dôvodom rozdelenia prevodu konečnostavového automatu na rekurentnú neurónovú sieť na dva typy. S výstupnou funkciou tento problém nenastáva, výstup konečnostavového automatu je binárny.

Pri riešení prevodu konečnosťavového automatu na rekurentnú neurónovú sieť sme pri automate s maximálne dvoma stavmi mohli využiť Karnaughovu mapu. Z tabuliek 3.2 a 3.3 vyplýva, že sa nejedná o logické funkcie, nakoľko reprezentácia je tvorená dvojicou číslíc, z ktorej prvá číslica môže nadobúdať hodnoty 0, 1 a 2 a spolu tvoria až 6 permutácií.

3.2 Tvorba neurónovej siete

Existuje niekoľko spôsobov, ako vytvoriť neurónovú sieť ekvivalentnú s konečnosťavovým automatom. V tejto časti popíšeme tri spôsoby, ktoré sme objavili a demonštrujeme ich priamo na príklade.

3.2.1 Výstupná a prechodová vrstva pre každý stav konečnosťavového automatu

Jednou z možností je vytvoriť samostatnú vrstvu obsahujúcu neuróny pre každý stav konečnosťavového automatu. Táto vrstva neurónov by v sebe zahŕňala prechodovú aj výstupnú funkciu stavu automatu. Keďže prechod v čase t je podmienený vstupným parametrom v čase t a prechodom v čase $t-1$ a možné prechody sa nedajú vyjadriť jedným binárnym číslom, vytvoríme pre každý prechod v konečnosťavovom automate samostatný neurón vo vrstve. To znamená, že pre každý stav v konečnosťavovom automate vytvoríme 3 neuróny. Jeden bude reprezentovať výstupnú funkciu, druhý bude reprezentovať prechodovú funkciu v prípade, že vstupným parametrom je hodnota 0 a tretí bude reprezentovať prechodovú funkciu v prípade, že vstupným parametrom je hodnota 1. Každý neurón, ktorý reprezentuje prechodovú funkciu bude vstupovať do vrstvy neurónov, ktoré reprezentujú výsledný stav danej prechodovej funkcie v konečnosťavovom automate.

Neurón reprezentujúci prechodovú funkciu stavu pri vstupnej hodnote 0, bude mať výstupnú hodnotu 1 iba v prípade, že vstupná hodnota bola 0 ($i^{(t)} = 0$) a zároveň daný stav bol aktivovaný predošlým stavom ($s^{(t)} = 1$).

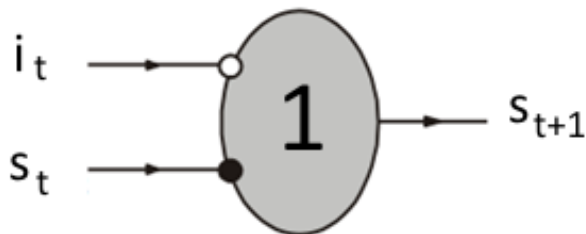
$i^{(t)} \setminus s^{(t)}$	0	1
0	0	1
1	0	0

Tabuľka 3.4: Karnaughova mapa pre prechodový neurón pri vstupnej hodnote 0

Túto funkciu už vieme zapísať pomocou Karnaughovej mapy (tabuľka 3.4) a teda vieme povedať, že neurón reprezentujúci prechodovú funkciu stavu pri vstupnej hodnote 0 má logický tvar:

$$\text{NOT } i^{(t)} \text{ AND } s^{(t)}$$

Túto logickú funkciu vieme jednoducho interpretovať neurónom na obrázku 3.2. V prípade, že do daného stavu môže vstúpiť prechod z viac ako jedného stavu, pridajú sa potrebné vstupy ako ďalšie exitačné vstupy do neurónu. Prah neurónu sa nemení nakoľko v čase t je aktívny vždy len jeden stav automatu.



Obr. 3.2: Neurón reprezentujúci prechodovú funkciu pri vstupnej hodnote 0

Neurón reprezentujúci prechodovú funkciu stavu pri vstupnej hodnote 1, bude mať výstupnú hodnotu 1 iba v prípade, že vstupná hodnota bola 1 ($i^{(t)} = 1$) a zároveň daný stav bol aktivovaný predošlým stavom ($s^{(t)} = 1$).

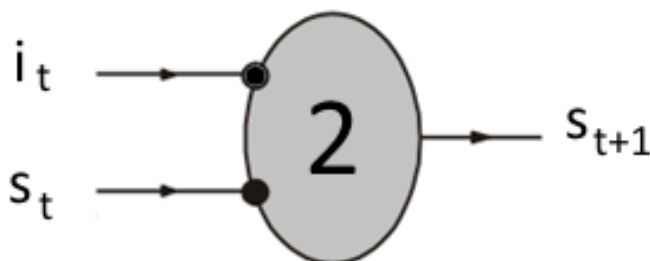
$i^{(t)} \setminus s^{(t)}$	0	1
0	0	0
1	0	1

Tabuľka 3.5: Karnaughova mapa pre prechodový neurón pri vstupnej hodnote 1

Túto funkciu vieme opäť zapísať pomocou Karnaughovej mapy (tabuľka 3.5) a teda neurón reprezentujúci prechodovú funkciu stavu pri vstupnej hodnote 1 má logický tvar:

$$i^{(t)} \text{ AND } s^{(t)}$$

Túto logickú funkciu vieme jednoducho interpretovať neurónom na obrázku 3.3. V prípade, že do daného stavu môže vstúpiť prechod z viac ako jedného stavu, pridajú sa potrebné vstupy ako ďalšie exitačné vstupy do neurónu. Prah neurónu sa nemení nakoľko v čase t je aktívny vždy len jeden stav automatu



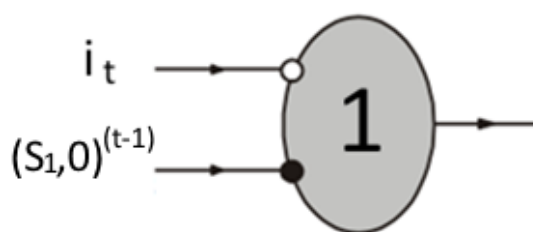
Obr. 3.3: Neurón reprezentujúci prechodovú funkciu pri vstupnej hodnote 1

Výstupný neurón pre každý stav konečnostavového neurónu sa tvorí na základe výstupnej funkcie každého stavu samostatne. Výstup pre stav S_0 pri vstupe 0 je 1 a pri vstupe 1 je 0. Výstup pre stav S_1 pri vstupe 0 je 0 a pri vstupe 1 je 0. Výstup pre stav S_2 pri vstupe 0 je 1 a pri vstupe 1 je 1. Výstupný neurón pre každý stav zároveň prijíma na vstup výstup prechodových neurónov všetkých stavov, ktoré môžu do daného stavu prejsť.

Do stavu S_0 konečnostavového automatu je možné prejsť len zo stavu S_1 pri vstupe 0. Preto výstupný neurón stavu S_0 bude mať dva vstupy: vstupnú hodnotu a stav prechodového neurónu pre stav S_1 automatu v prípade vstupu s hodnotou 0. Označme si stav prechodového neurónu pre stav S_1 pri vstupe 0 vstupujúci do výstupného neurónu stavu S_0 ako $(S_1, 0)^{(t-1)}$. Potom Karnaughova mapa pre výstupný neurón stavu S_0 bude zodpovedať hodnotám v tabuľke 3.6 a neurón bude korešpondovať s obrázkom 3.4.

$i^{(t)} \setminus (S_1, 0)^{(t-1)}$	0	1
0	0	1
1	0	0

Tabuľka 3.6: Karnaughova mapa pre výstupný neurón stavu S_0



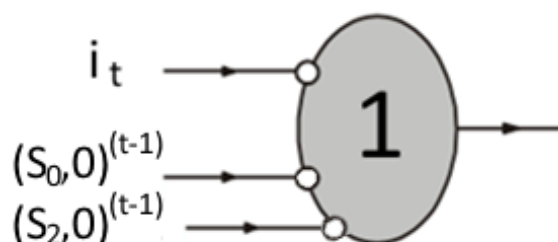
Obr. 3.4: Výstupný neurón pre stav S_0

Do stavu S_1 konečnostavového automatu je možné prejsť zo stavu S_0 pri vstupe 0 alebo zo stavu S_2 pri vstupe 0. Preto výstupný neurón stavu S_1 bude mať tri vstupy: vstupnú hodnotu, stav $(S_0, 0)^{(t-1)}$ prechodového neurónu pre stav S_0 automatu v prípade vstupu s hodnotou 0 a stav $(S_2, 0)^{(t-1)}$ prechodového neurónu pre stav S_2 automatu v prípade vstupu s hodnotou 0. Potom Karnaughova mapa pre výstupný neurón stavu S_1 bude zodpovedať hodnotám v tabuľke 3.7 (pozn.: tretí stĺpec hodnôt, teda možnosť kedy $(S_0, 0)^{(t-1)}$ sa rovná 1 a zároveň $(S_2, 0)^{(t-1)}$ sa rovná 1, nikdy nenastane nakoľko je vždy v akomkoľvek čase aktívny len jeden prechod). Z tabuľky 3.7 vyplýva, že výstupný neurón

pre stav S_1 konečnostavového automatu bude mať vždy výstup 0 z čoho vyplýva, že výstupný neurón bude korešpondovať s neurónom na obrázku 3.5.

	$\overline{(S_0, 0)^{(t-1)}, (S_2, 0)^{(t-1)}}$	$(S_0, 0)^{(t-1)}, \overline{(S_2, 0)^{(t-1)}}$	$(S_0, 0)^{(t-1)}, (S_2, 0)^{(t-1)}$	$\overline{(S_0, 0)^{(t-1)}, (S_2, 0)^{(t-1)}}$
$\overline{i^{(t)}}$	0	0	X	0
$i^{(t)}$	0	0	X	0

Tabuľka 3.7: Karnaughova mapa pre výstupný neurón stavu S_1



Obr. 3.5: Výstupný neurón pre stav S_1

Do stavu S_2 konečnostavového automatu je možné prejsť zo stavu S_0 pri vstupe 1, zo stavu S_1 pri vstupe 1 alebo zo stavu S_2 pri vstupe 1. Preto výstupný neurón stavu S_2 bude mať štyri vstupy: vstupnú hodnotu, stav $(S_0, 1)^{(t-1)}$ prechodového neurónu pre stav S_0 automatu v prípade vstupu s hodnotou 1, stav $(S_1, 1)^{(t-1)}$ prechodového neurónu pre stav S_1 automatu v prípade vstupu s hodnotou 1 a stav $(S_2, 1)^{(t-1)}$ prechodového neurónu pre stav S_2 automatu v prípade vstupu s hodnotou 1. Potom Karnaughova mapa pre výstupný neurón stavu S_2 bude zodpovedať hodnotám v tabuľke 3.8 (pozn.: tretí stĺpec hodnôt, teda možnosť kedy $(S_0, 1)^{(t-1)}$ sa rovná 1 a zároveň $(S_2, 1)^{(t-1)}$ sa rovná 1 nikdy nenastane nakoľko je vždy v akomkoľvek čase aktívny len jeden prechod. Zároveň nikdy nenastane stav tretieho a štvrtého riadku druhého a štvrtého stĺpca, teda možnosť kedy sa $(S_1, 1)^{(t-1)}$ rovná 1 a zároveň sa $(S_0, 1)^{(t-1)}$ alebo $(S_2, 1)^{(t-1)}$ rovná 1). Z tabuľky 3.8 vyplýva, že výstupný neurón pre stav S_2 konečnostavového automatu bude mať vždy výstup 1 ak bol aktivovaný predošlými stavmi. Zároveň z tabuľky 3.8 vyplýva, že pre

interpretáciu tejto Karnaughovej mapy neurónom budeme potrebovať viac ako 1 neurón, nakoľko výstupná funkcia má v logickej interpretácii tvar:

$$((S_0, 1)^{(t-1)}, \overline{(S_2, 1)^{(t-1)}}), (S_1, 1)^{(t-1)}) \text{ OR } (\overline{(S_0, 1)^{(t-1)}}, \overline{(S_2, 1)^{(t-1)}}), (S_1, 1)^{(t-1)}) \text{ OR } (\overline{(S_0, 1)^{(t-1)}}, (S_2, 1)^{(t-1)}, \overline{(S_1, 1)^{(t-1)})})$$

$$\overline{(S_0, 1)^{(t-1)}, (S_2, 1)^{(t-1)}} (S_0, 1)^{(t-1)}, \overline{(S_2, 1)^{(t-1)}} (S_0, 1)^{(t-1)}, (S_2, 1)^{(t-1)} \overline{(S_0, 1)^{(t-1)}, (S_2, 1)^{(t-1)}}$$

$\overline{i^{(t)}}, \overline{(S_1, 1)^{(t-1)}}$	0	1	X	1
$i^{(t)}, \overline{(S_1, 1)^{(t-1)}}$	0	1	X	1
$i^{(t)}, (S_1, 1)^{(t-1)}$	1	X	X	X
$\overline{i^{(t)}}, (S_1, 1)^{(t-1)}$	1	X	X	X

Tabuľka 3.8: Karnaughova mapa pre výstupný neurón stavu S_2

Ak by sme ale Karnaughovu mapu upravili na základe znalosti, že nenastane moment, kedy by dva prechodové stavy boli aktívne v ten istý čas, mohli by sme docieľiť zjednodušenie logickej reprezentácie výstupovej funkcie pre neurón. Doplnením jednotkových stavov do druhého a štvrtého stĺpca v tabuľke 3.8 vznikne nová Karnaughova mapa v tabuľke 3.9.

$$\overline{(S_0, 1)^{(t-1)}, (S_2, 1)^{(t-1)}} (S_0, 1)^{(t-1)}, \overline{(S_2, 1)^{(t-1)}} (S_0, 1)^{(t-1)}, (S_2, 1)^{(t-1)} \overline{(S_0, 1)^{(t-1)}, (S_2, 1)^{(t-1)}}$$

$\overline{i^{(t)}}, \overline{(S_1, 1)^{(t-1)}}$	0	1	X	1
$i^{(t)}, \overline{(S_1, 1)^{(t-1)}}$	0	1	X	1
$i^{(t)}, (S_1, 1)^{(t-1)}$	1	1	X	1
$\overline{i^{(t)}}, (S_1, 1)^{(t-1)}$	1	1	X	1

Tabuľka 3.9: Upravená Karnaughova mapa pre výstupný neurón stavu S_2

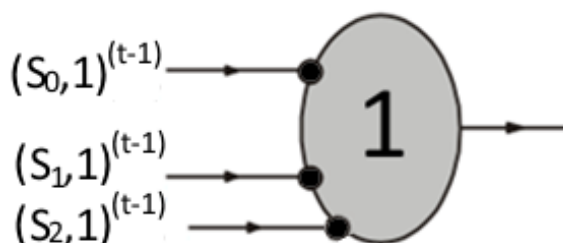
Logická interpretácia upravenej Karnaughovej mapy z tabuľky 3.9 má tvar:

$$((S_0, 1)^{(t-1)}, \overline{(S_2, 1)^{(t-1)}}) \text{ OR } (\overline{(S_0, 1)^{(t-1)}}, (S_2, 1)^{(t-1)}) \text{ OR } ((S_1, 1)^{(t-1)}, \overline{(S_2, 1)^{(t-1)})}$$

Z logickej interpretácie vyplýva, že pre výstupovú funkciu stavu S_2 konečnostavového automatu nepotrebujeme poznať vstupný parameter. Na základe týchto poznatkov môžeme povedať, že výstupný neurón stavu S_2 bude mať tri vstupy: stav $(S_0, 1)^{(t-1)}$ prechodového neurónu pre stav S_0 automatu v prípade vstupu s hodnotou 1, stav $(S_1, 1)^{(t-1)}$ prechodového neurónu pre stav S_1 automatu v prípade vstupu s hodnotou 1 a stav $(S_2, 1)^{(t-1)}$ prechodového neurónu pre stav S_2 automatu v prípade vstupu s hodnotou 1. Po aplikovaní všetkých poznatkov môžete logickú interpretáciu upraviť na tvar (funkcia OR):

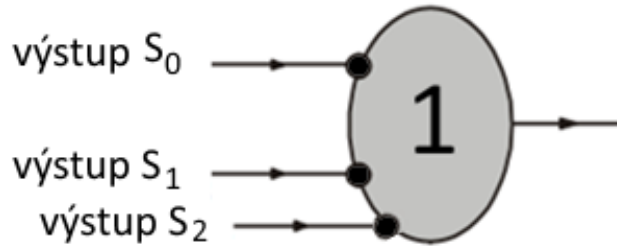
$$((S_0, 1)^{(t-1)}) \text{ OR } ((S_2, 1)^{(t-1)}) \text{ OR } ((S_1, 1)^{(t-1)})$$

Výstupný neurón reprezentujúci výstupnú funkciu stavu S_2 konečnostavového automatu bude korešpondovať s neurónom na obrázku 3.6.



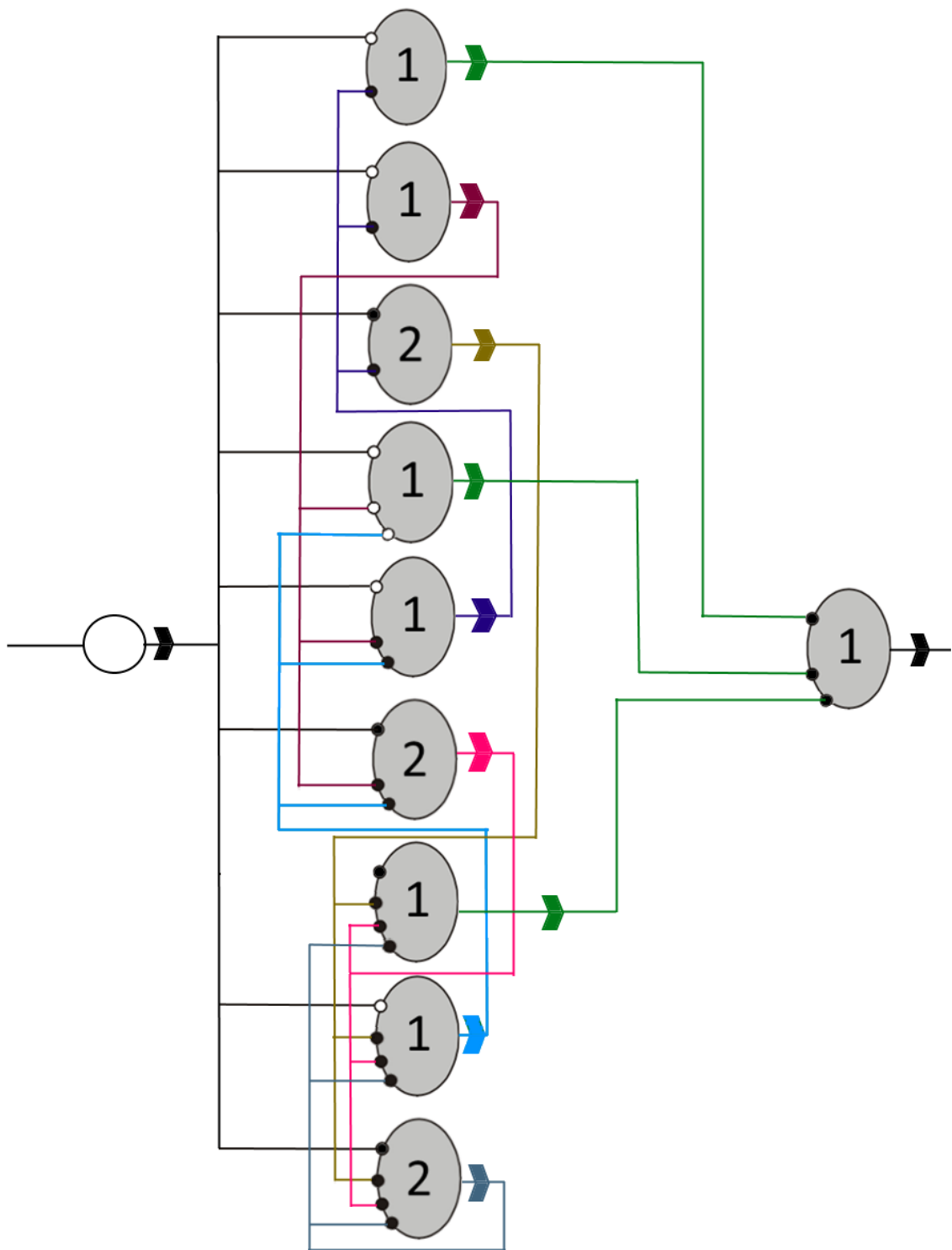
Obr. 3.6: Výstupný neurón pre stav S_2

Po spojení všetkých vyššie uvedených neurónov do neurónovej siete vznikne sieť s tromi výstupmi v čase. Preto treba výstupy z výstupných neurónov spojiť do jedného výstupu. V prípade, že daný stav nebol aktivovaný, resp. všetky prechodové neuróny vstupujúce ho výstupného neurónu boli nulové, potom aj výstup výstupného neurónu reprezentujúceho tejto stav je nulový. Tento neurón bude mať tým pádom logickú reprezentáciu funkcie OR. Neurón, ktorý spája výstupy z výstupných neurónov pre stavy S_0, S_1 a S_2 do jedného konečného výstupu je znázornený na obrázku 3.7.

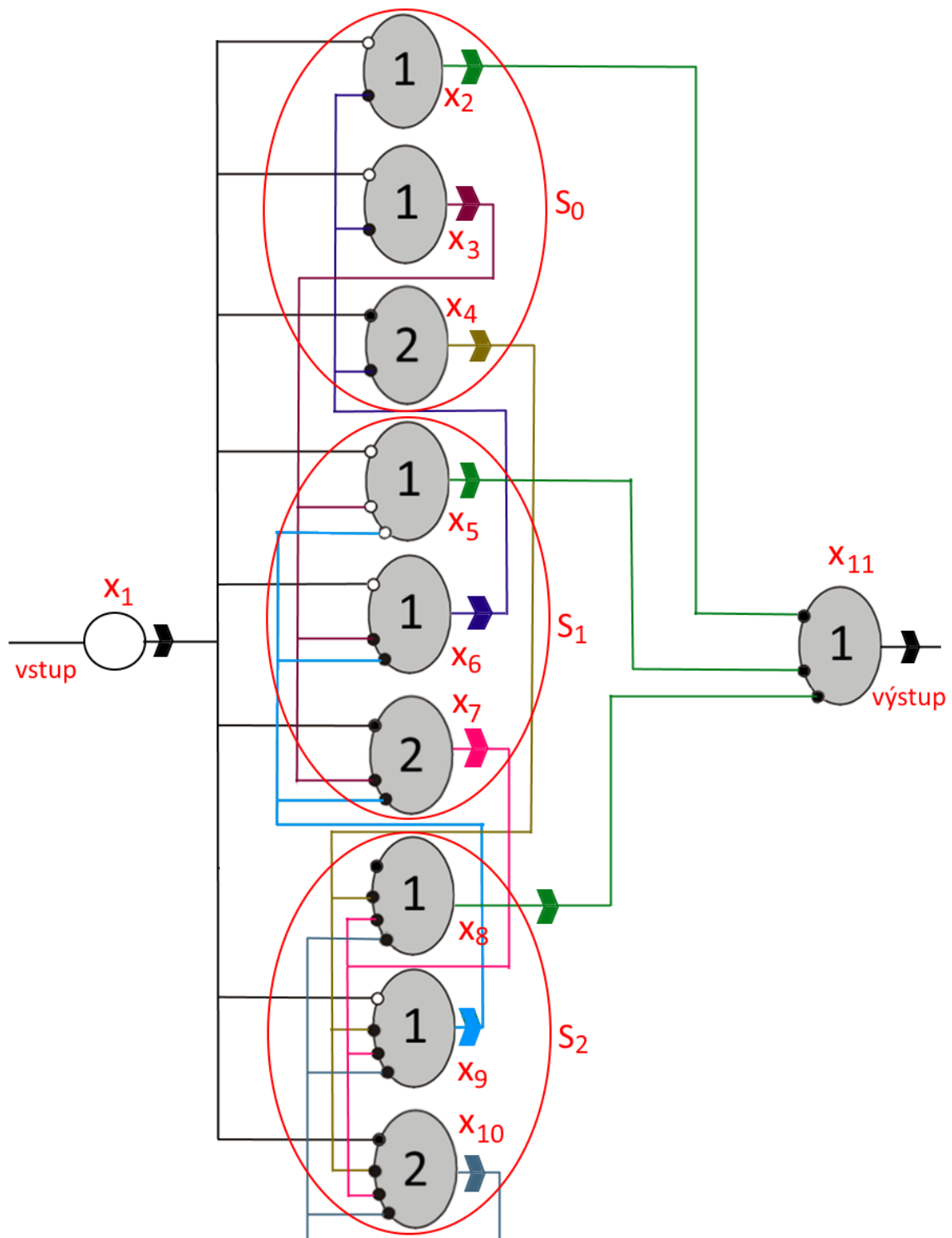


Obr. 3.7: Výstupný neurón rekurentnej neurónovej siete

Výsledná rekurentná neurónová sieť ekvivalentná s konečnosťavým automatom na obrázku 3.1 je zobrazená na obrázku 3.8. Každý stav konečnosťavového automatu je reprezentovaný trojicou neurónov, do ktorých vstupujú vstupné dáta (pozn.: v tomto prípade s výnimkou výstupného neurónu posledného stavu kde sme skonštatovali, že odstránením vstupných dát v tomto neuróne zachováme a zároveň zjednodušíme výstupnú funkciu) a výstupy prechodových funkcií, ktoré sú rozdelené na základe vstupu 0 alebo 1 a ktorých úlohou je aktivovať nasledujúci stav, resp. skupinu neurónov zodpovedajúcu danému stavu v neurónovej sieti.



Obr. 3.8: Rekurentná neurónová sieť ekvivalentná so zadaným konečnosťovým automatom



Obr.: 3.9: Rekurentná neurónová sieť s popismi

Obrázok 3.9 zobrazuje rekurentnú neurónovú sieť z obrázka 3.8 s dodatočnými popismi. Každá trojica neurónov, ktorá reprezentuje jeden stav v konečnostavovom automate, sa skladá z výstupného neurónu a dvoch prechodových neurónov. Všetky

výstupné neuróny posielajú výstup do konečného výstupu. V prípade, že daný stav (resp. trojica neurónov) nebol aktivovaný v danom čase, výstup daného stavu (resp. výstup výstupného neurónu) je nulový. Stav (resp. trojica neurónov) je aktivovaný len v prípade, ak jeden z výstupov prechodových neurónov doňho smerujúcich sa rovná 1. Z tohto dôvodu musíme pri testovaní ekvivalencie výpočtom stavov neurónov v čase nastaviť v počiatočnom momente stav prechodového neurónu zodpovedajúceho stavu S_1 pri vstupe 0 na 1, nakoľko tento neurón aktivuje začiatkový stav S_0 .

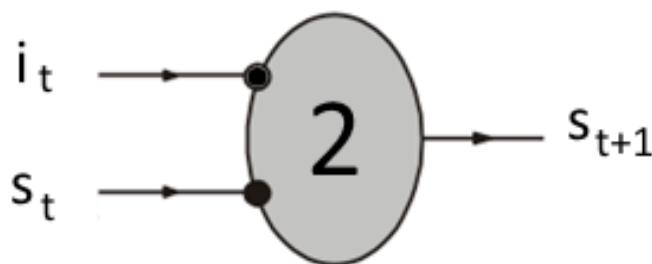
Pre otestovanie ekvivalencie rekurentnej neurónovej siete s konečnostavovým automatom vypočítame pre každý z nich výstupný reťazec pri tom istom vstupnom reťazci. Pre vstupný reťazec (001100010) sa výstup konečnostavového automatu na obrázku 3.1 rovná postupnosti (100110101). Prechodová postupnosť sa rovná postupnosti ($S_1, S_0, S_2, S_2, S_1, S_0, S_1, S_2, S_1$). Rekurentnú neurónovú sieť z obrázka 3.8 otestujeme so vstupným reťazcom (001100010) a porovnáme s výstupom, ktorý vygeneroval konečnostavový automat na obrázku 3.1 s prechodovou a výstupnou funkciou v tabuľke 3.1. Výsledné stavy neurónov v čase tejto rekurentnej neurónovej siete sú v tabuľke 3.10.

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}
-	0	0	0	0	1	0	0	0	0	-
0	1	1	0	0	0	0	0	0	0	1
0	0	0	0	0	1	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	1	0	1	1
0	0	0	0	0	0	0	1	1	0	1
0	0	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	0	0	0	0	1
1	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	1	1	0	1

Tabuľka 3.10: Stavy neurónov v čase

Výstupný reťazec $x_{11} = (100110101)$ korešponduje s výsledným reťazcom vygenerovaným konečnosťovým automatom. V tabuľke 3.10 môžeme overiť aj zhodu prechodovej postupnosti. Neuróny x_3 a x_4 reprezentujú stav S_0 . Neuróny x_6 a x_7 reprezentujú stav S_1 . Neuróny x_9 a x_{10} reprezentujú stav S_2 . V prípade, že jeden z dvojice neurónov mal v čase t stav rovný 1, tak daný stav bol v čase t aktívny. Z toho môžeme vyvodit' prechodovú postupnosť $(S_1, S_0, S_2, S_2, S_1, S_0, S_1, S_2, S_1)$. Nultý čas, v ktorom neurón x_6 má stav 1, ignorujeme.

V testovanom konečnosťovom automate sme mali tri stavy, z ktorých každý generoval rôzny výstup. Pri binárnom výstupe 0 alebo 1 máme však 4 rôzne kombinácie (00, 01, 10 a 11). Štvrtou chýbajúcou kombináciou je výstup 0 pri vstupe 0 a výstup 1 pri vstupe 1, ktorý sa v testovanom konečnosťovom automate nenachádza. Jeho výstupový neurón je zobrazený na obrázku 3.10. Každý ďalší vstup z prechodových funkcií sa pridáva ako exitačný vstup do neurónu.



Obr. 3.10: Výstupný neurón pre štvrtý typ stavu

Na základe vyššie popísaného postupu vyplýva, že týmto spôsobom vieme vytvorit' rekurentnú neurónovú sieť ekvivalentnú s konečnosťovým automatom a pre každý stav potrebujeme vytvorit' 3 neuróny a pre konečný výstup potrebujeme jeden neurón s exitačnými vstupmi zo všetkých výstupných neurónov každého stavu a s prahom 1. Počet neurónov potrebných pre vytvorenie rekurentnej neurónovej siete ekvivalentnej s konečnosťovým automatom vieme vyjadrit' funkciou:

$$n = 3 * S + 1$$

kde:

- n je počet neurónov
- S je počet stavov konečnostavového automatu

Nesmieme však zabudnúť vždy na začiatku testovania neurónovej siete aktivovať prechodový neurón v čase t , ktorý vstupuje v čase $t+1$ do začiatočného stavu. V prípade, že neexistuje stav vracajúci sa do začiatočného stavu, je potrebné vytvoriť umelý neurón s počiatočným stavom 1, ktorý vstupuje do začiatočného neurónu a aktivuje začiatočný stav pri štarte neurónovej siete.

3.2.2 Prechodová vrstva pre každý stav konečnostavového automatu a samostatná výstupná funkcia

V kapitole 3.2.1 sme popísali a overili postup vytvorenia rekurentnej neurónovej siete ekvivalentnej konečnostavového automatu pomocou vytvorenia výstupovej a prechodovej funkcie reprezentovaných neurónmi pre každý stav automatu samostatne. Ak ale poznáme vstupný parameter a zároveň vieme, v ktorom stave sa automat nachádza, vieme určiť výstup.

Na základe tohto poznatku môžeme upraviť postup v kapitole 3.2.1 a spojiť výstupné neuróny každého stavu do jedného celku. To znamená, že v skrytej vrstve rekurentnej neurónovej siete ostanú neuróny reprezentujúce prechodovú funkciu (neuróny na obrázku 3.2 a 3.3) a výstupné neuróny sa spoja do jedného celku tvoreného jedným alebo viacerými neurónmi v závislosti od výstupnej funkcie konečnostavového automatu.

Výstup závisí od aktuálneho vstupu a aktuálne aktivovaného stavu. Opäť je samozrejmé, že v akomkoľvek čase je aktivovaný iba jeden stav. To znamená, že môžu nastať len situácie zobrazené v tabuľke 3.3. Vo výstupnej funkcii nezáleží na tom, do akého stavu daný prechodový neurón smeruje. Záleží na tom, ktorý stav bol v danom čase aktívny. Preto môžeme tieto dvojice prechodových neurónov v Karnaughovej mape spojiť. Hodnoty výstupnej funkcie vieme zobraziť v Karnaughovej mape v tabuľke 3.11, kde i_t reprezentuje vstupnú hodnotu v čase t , S_0 reprezentuje prechodové neuróny $(S_0, 0)^{(t)}$ a $(S_0, 1)^{(t)}$, S_1 reprezentuje prechodové neuróny $(S_1, 0)^{(t)}$ a $(S_1, 1)^{(t)}$ a S_2 reprezentuje prechodové neuróny $(S_2, 0)^{(t)}$ a $(S_2, 1)^{(t)}$.

	\bar{i}_t, \bar{S}_0	i_t, \bar{S}_0	i_t, S_0	\bar{i}_t, S_0
\bar{S}_1, \bar{S}_2	X	X	0	1
S_1, \bar{S}_2	0	0	X	X
S_1, S_2	X	X	X	X
\bar{S}_1, S_2	1	1	X	X

Tabuľka 3.11: Karnaughova mapa pre výstupnú funkciu

Hodnota X v tabuľke 3.11 reprezentuje možnosti, ktoré nikdy nenastanú – to znamená, že vždy bude jeden stav aktívny (stav jedného z prechodových neurónov bude rovný 1). Logická reprezentácia Karnaughovej mapy v tabuľke 3.11 má tvar:

$$\bar{S}_1, S_2, \bar{S}_0 \text{ OR } \bar{i}_t, S_0, \bar{S}_1, \bar{S}_2$$

Rovnako ako pri výstupnom neuróne pre stav S_2 na obrázku 3.6, aj tu si môžeme pomôcť úpravou Karnaughovej mapy za účelom zjednodušenia logickej reprezentácie.

	\bar{i}_t, \bar{S}_0	i_t, \bar{S}_0	i_t, S_0	\bar{i}_t, S_0
\bar{S}_1, \bar{S}_2	X	X	0	1
S_1, \bar{S}_2	0	0	X	1
S_1, S_2	1	1	1	1
\bar{S}_1, S_2	1	1	1	1

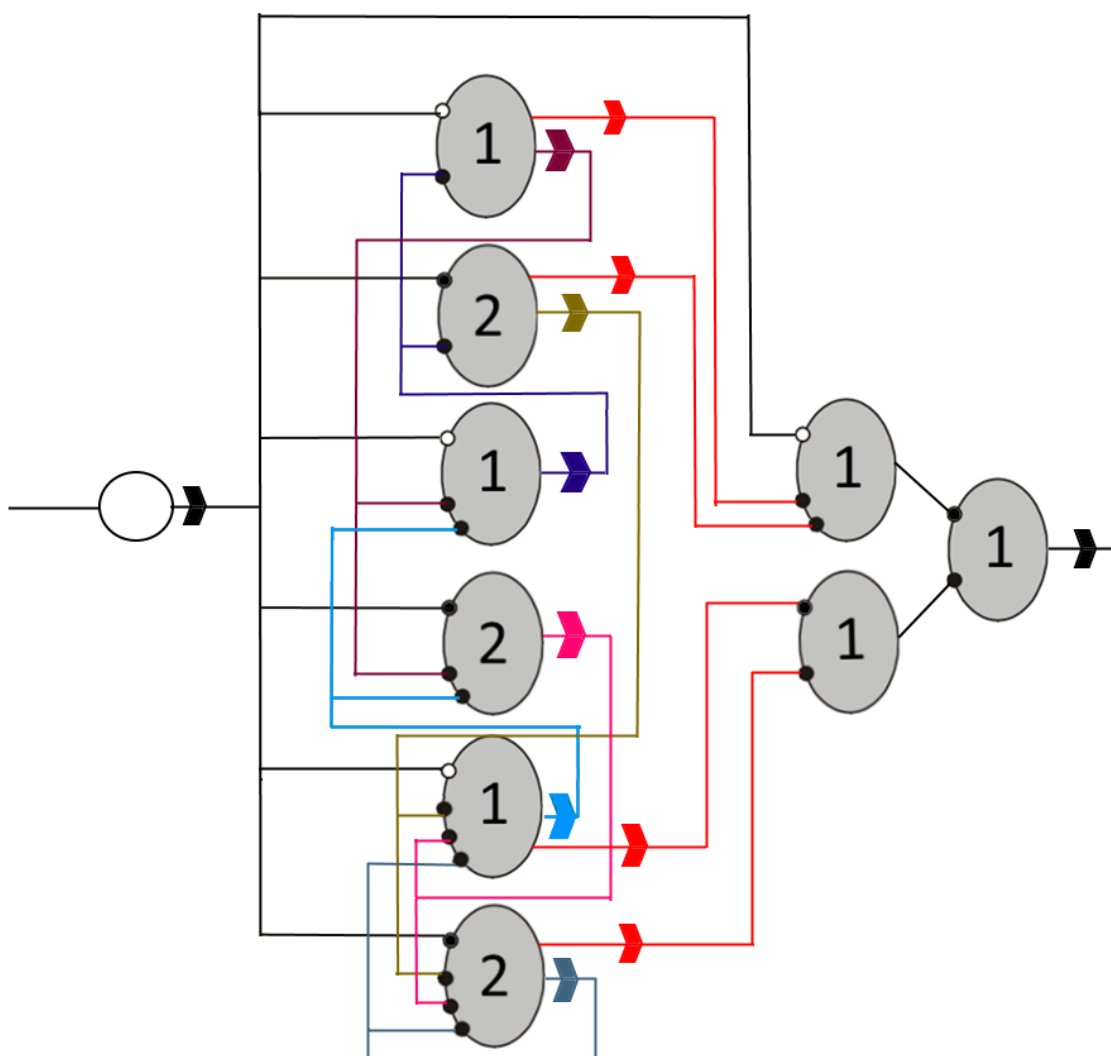
Tabuľka 3.12: Upravená Karnaughova mapa pre výstupnú funkciu

Po úprave má logická reprezentácia výstupnej funkcie tvar:

$$S_2 \text{ OR } \bar{i}_t, S_0$$

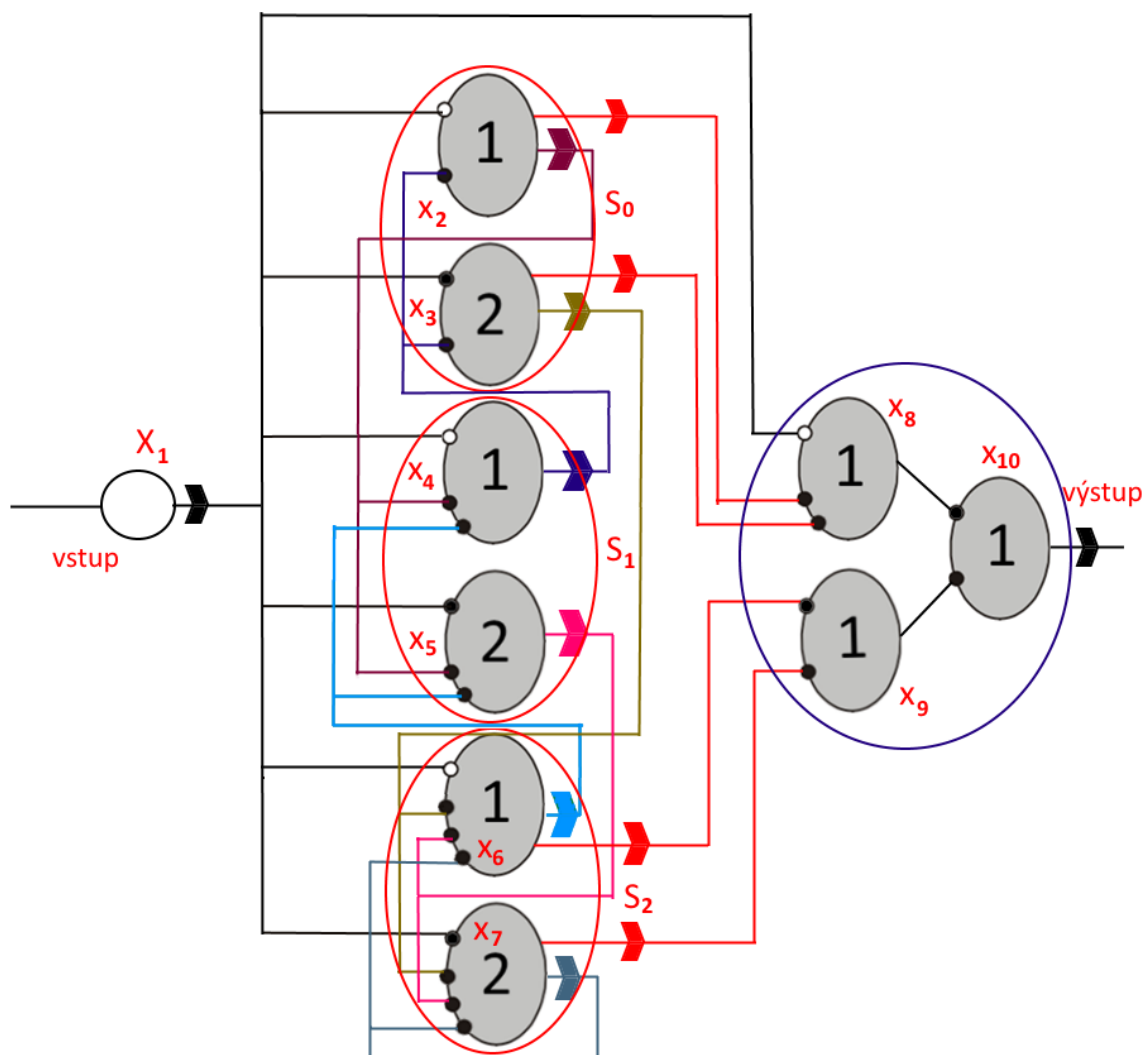
Tvar výstupnej funkcie naznačuje, že nepotrebujeme prepojiť prechodové neuróny stavu S_1 s výstupnou funkciou. Zároveň naznačuje, že pre interpretáciu tejto funkcie budeme potrebovať viac ako jeden neurón, nakoľko nie je možné túto logickú reprezentáciu interpretovať jedným neurónom aj napriek znalostiach, že nikdy nie sú aktívne 2 stavy v jednom čase.

Rekurentná neurónová sieť ekvivalentná s konečnosťavým automatom na obrázku 3.1 je zobrazená na obrázku 3.11.



Obr. 3.11: Rekurentná neurónová sieť ekvivalentná so zadaným konečnosťavým automatom

Každý stav je reprezentovaný dvojicou prechodových neurónov a výstupná funkcia je vytvorená neurónmi reprezentujúcimi logickú reprezentáciu Karnaughovej mapy v tabuľke 3.12.



Obr. 3.12: Rekurentná neurónová sieť s popismi

Tento spôsob prevodu sa približuje k schematickej reprezentácii konečnostavového automatu reprezentovaného prechodovou a výstupnou funkciou na obrázku 2.2.

Rekurentnú neurónovú sieť z obrázka 3.11 otestujeme so vstupným reťazcom (001100010) a porovnáme s výstupom, ktorý vygeneroval konečnostavový automat na

obrázku 3.1 s prechodovou a výstupnou funkciou v tabuľke 3.1. Pri testovaní ekvivalencie výpočtom stavov neurónov v čase musíme nastaviť v počiatočnom momente stav prechodového neurónu zodpovedajúcemu stavu S_1 pri vstupe 0 na 1, nakoľko tento neurón aktivuje začiatočný stav S_0 ako už bolo vysvetlené v kapitole 3.2.1.

Výsledné stavy neurónov v čase tejto rekurentnej neurónovej siete sú v tabuľke 3.13.

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
-	0	0	1	0	0	0	0	0	-
0	1	0	0	0	0	0	1	0	1
0	0	0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0	0
1	0	0	0	0	0	1	0	1	1
0	0	0	0	0	1	0	0	1	1
0	0	0	1	0	0	0	0	0	0
0	1	0	0	0	0	0	1	0	1
1	0	0	0	1	0	0	0	0	0
0	0	0	0	0	1	0	0	1	1

Tabuľka 3.13: Stavy neurónov v čase

Výstupný reťazec $x_{11} = (100110101)$ korešponduje s výsledným reťazcom vygenerovaným konečnostavovým automatom. V tabuľke 3.13 môžeme overiť aj zhodu prechodovej postupnosti. Neuróny x_2 a x_3 reprezentujú stav S_0 . Neuróny x_4 a x_5 reprezentujú stav S_1 . Neuróny x_6 a x_7 reprezentujú stav S_2 . V prípade, že jeden z dvojice neurónov mal v čase t stav rovný 1, tak daný stav bol v čase t aktívny. Z toho môžeme vyvodíť prechodovú postupnosť $(S_1, S_0, S_2, S_2, S_1, S_0, S_1, S_2, S_1)$. Nultý čas, v ktorom neurón x_4 má stav 1, ignorujeme.

Na základe vyššie popísaného postupu vyplýva, že týmto spôsobom vieme vytvoriť rekurentnú neurónovú sieť ekvivalentnú s konečnosťavým automatom a pre každý stav potrebujeme vytvoriť 2 prechodové neuróny a výstupnú funkciu musíme interpretovať jej logickou reprezentáciou. Nesmieme však zabudnúť vždy na začiatku testovania neurónovej siete aktivovať prechodový neurón v čase t , ktorý vstupuje v čase $t+1$ do začiatočného stavu. V prípade, že neexistuje stav vracajúci sa do začiatočného stavu, je potrebné vytvoriť umelý neurón s počiatočným stavom 1, ktorý vstupuje do začiatočného neurónu a aktivuje začiatočný stav pri štarte neurónovej siete.

Maximálny počet neurónov potrebných pre vytvorenie rekurentnej neurónovej siete ekvivalentnej s konečnosťavým automatom vieme vyjadriť funkciou:

$$\max n = 2 * S + S + 1$$

$$\max n = 3 * S + 1$$

kde:

- n je počet neurónov
- S je počet stavov konečnosťavového automatu

Maximálny počet neurónov potrebných pre vytvorenie výstupnej funkcie je rovný počtu neurónov a jednému konečnému neurónu reprezentujúceho logickú operáciu OR. Z príkladu, ktorý je vyššie opísaný ale vyplýva, že pre tento konkrétny prípad sme pre výstupnú funkciu potrebovali len 3 neuróny, nakoľko vo výstupnej funkcii nezáležalo na stave prechodových neurónov reprezentujúcich stav S_1 automatu. Taktiež môže nastať situácia, kedy bude možné výstupnú funkciu reprezentovať jediným neurónom.

3.2.3 Prechodová vrstva pre každý stav konečnosťavového automatu a ich použitie pre priamy výstup

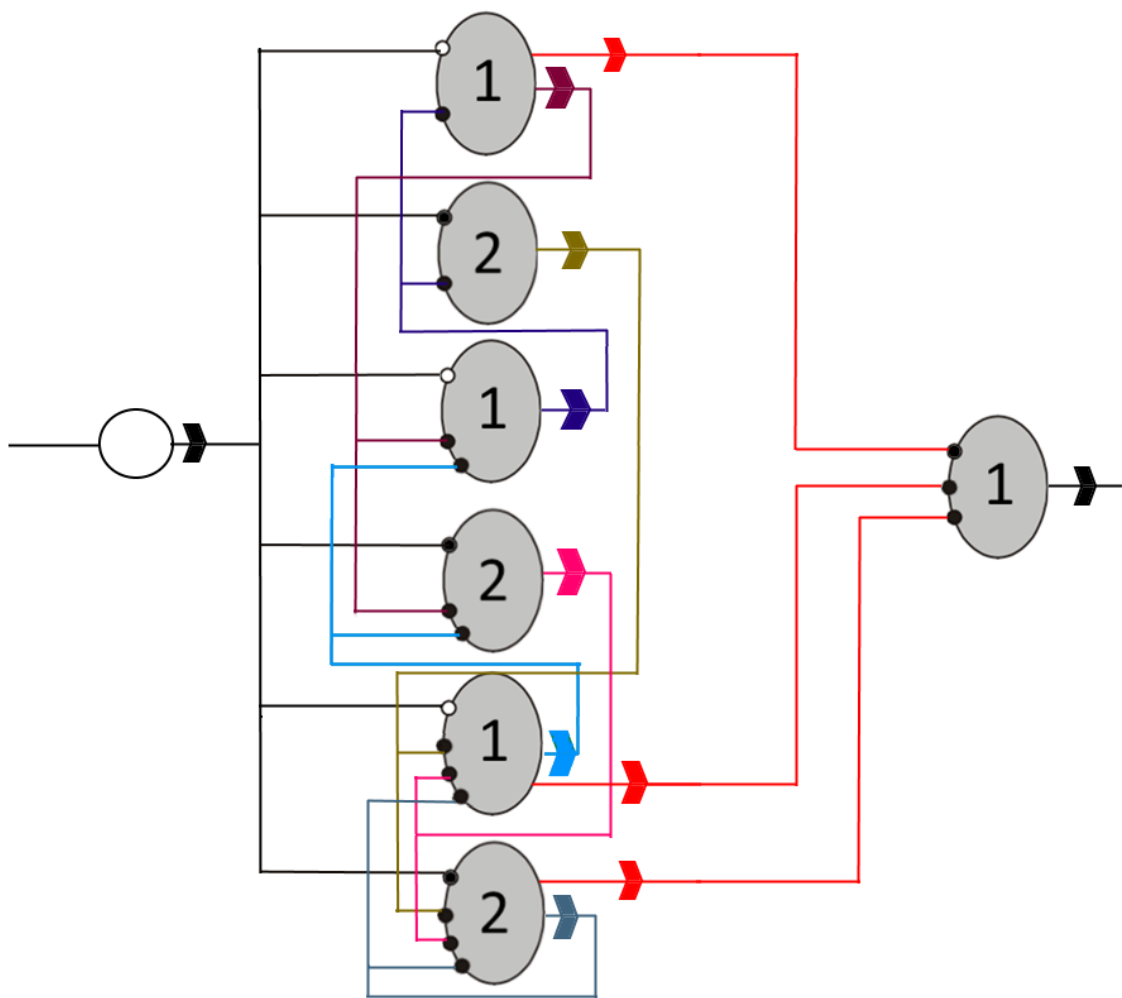
V kapitole 3.2.1 sme popísali a overili postup vytvorenia rekurentnej neurónovej siete ekvivalentnej konečnosťavového automatu pomocou vytvorenia výstupovej a prechodovej funkcie reprezentovaných neurónmi pre každý stav automatu samostatne. V kapitole 3.2.2 sme popísali a overili postup vytvorenia siete pomocou prechodovej funkcie

reprezentovaných neurónmi pre každý stav automatu samostatne a samostatnej výstupnej neurónovej vrstvy. V treťom spôsobe prevodu konečnostavového automatu na rekurentnú neurónovú sieť využijeme informáciu o aktivácii konkrétneho prechodu spolu s jeho výstupnou hodnotou.

Spôsob, akým reprezentujeme jednotlivé prechody automatu v neurónovej sieti (v kapitole 3.2.1 a 3.2.2), nám umožňuje presne identifikovať kedy tieto prechody nastávajú. Každý prechod konečnostavového automatu je reprezentovaný samostatným neurónom v sieti a jeho aktiváciu alebo neaktiváciu určujeme jeho stavom v čase. Napríklad: ak stav prechodového neurónu reprezentujúceho stav S_0 pri vstupe 0 má hodnotu 1 znamená to, že v danom čase sa automat nachádza v stave S_0 a na vstupe prišla hodnota 0. Pokiaľ túto skutočnosť spojíme so znalosťou výstupných hodnôt jednotlivých prechodov, môžeme úplne vynechať logickú reprezentáciu výstupnej funkcie a výstup realizovať priamo z prechodového neurónu. Z obrázka 3.1 konečnostavového automatu ako aj z tabuľky 3.1 prechodovej a výstupnej funkcie konečnostavového automatu vyplýva, že výstup s hodnotou 1 nastáva len pri stave S_0 a vstupe 0, pri stave S_2 a vstupe 0 a pri stave S_2 a vstupe 1. Zároveň stavy prechodových neurónov týchto prechodov automatu majú pri aktivácii hodnotu 1.

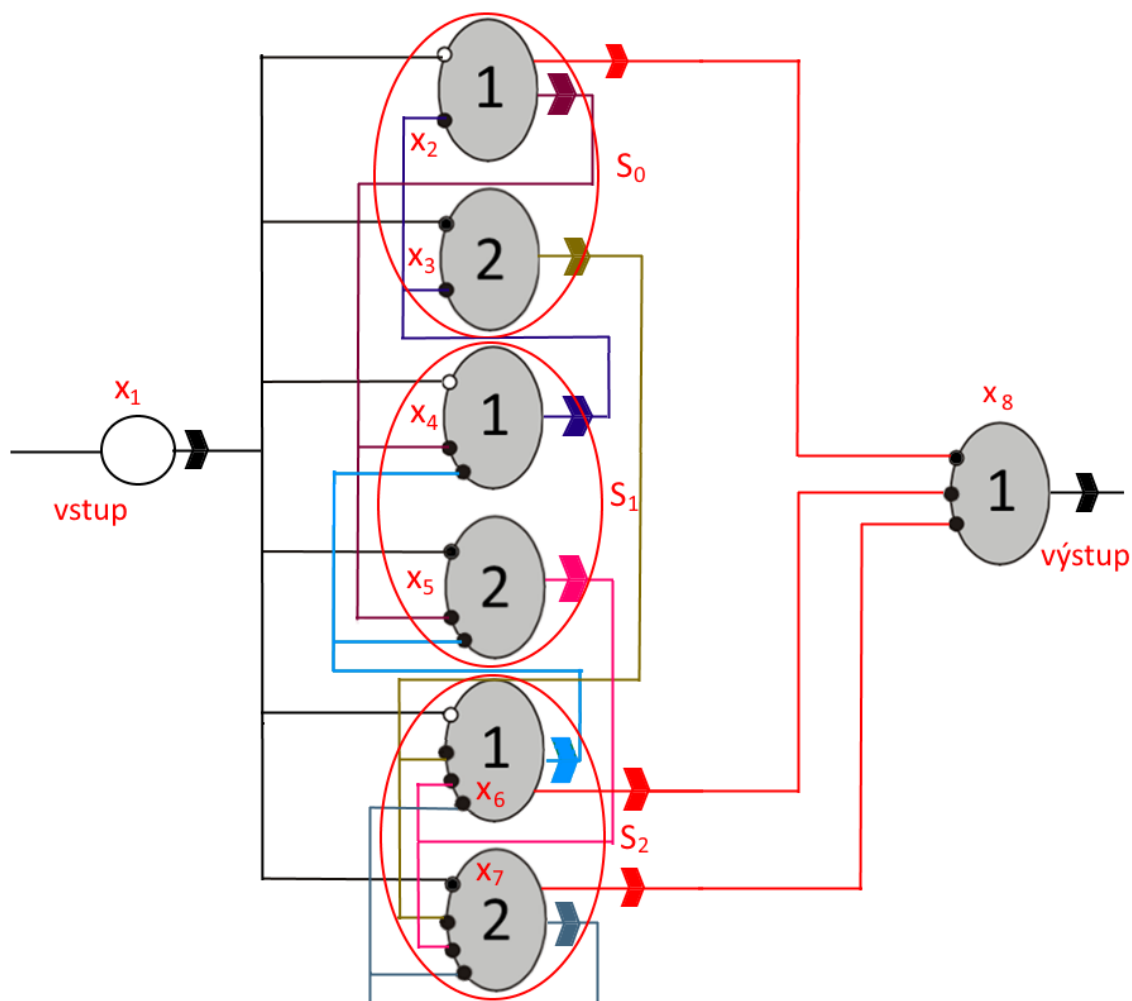
Pre výstup neurónovej siete ekvivalentnej konečnostavovému automatu na základe vyššie popísaných skutočností môžeme použiť výstup prechodových neurónov. Pre zjednodušenie neurónovej siete prepojíme výstupný neurón spájajúci výstupy prechodových neurónov len s tými prechodovými neurónmi, ktorých výstup má hodnotu 1 pri jeho aktivácii. Pri aplikovaní tohto postupu na konečnostavový automat z obrázka 3.1 bude v neurónovej sieti do výstupného neurónu vchádzať na exitačnom vstupe výstup prechodových neurónov $(S_0, 0)$, $(S_2, 0)$ a $(S_2, 1)$ a výstupný neurón bude mať prah 1 (funkcia OR).

Rekurentná neurónová sieť ekvivalentná s konečnostavovým automatom na obrázku 3.1 zostrojená vyššie opísaným postupom je zobrazená na obrázku 3.13.



Obr. 3.13: Rekurentná neurónová sieť ekvivalentná so zadaným konečnosťavým automatom

Týmto postupom sme eliminovali potrebu logickej reprezentácie výstupnej funkcie a priamo využívame znalosti o výstupe konkrétneho prechodu. Z obrázku 3.13 jasne vyplýva, že týmto postupom radikálne eliminujeme počet neurónov potrebných pre zostrojenie rekurentnej neurónovej siete ekvivalentnej s konečnosťavým automatom.



Obr. 3.14: Rekurentná neurónová sieť s popismi

Rekurentnú neurónovú sieť z obrázka 3.13 otestujeme so vstupným reťazcom (001100010) a porovnáme s výstupom (100110101), ktorý vygeneroval konečnostavový automat na obrázku 3.1 s prechodovou a výstupnou funkciou v tabuľke 3.1. Pri testovaní ekvivalencie výpočtom stavov neurónov v čase musíme opäť nastaviť v počiatocnom momente stav prechodového neurónu zodpovedajúceho stavu S_1 pri vstupe 0 na 1, nakoľko tento neurón aktivuje začiatocný stav S_0 ako už bolo vysvetlené v kapitole 3.2.1.

Výsledné stavy neurónov v čase tejto rekurentnej neurónovej siete sú v tabuľke 3.14.

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
-	0	0	1	0	0	0	-
0	1	0	0	0	0	0	1
0	0	0	1	0	0	0	0
1	0	1	0	0	0	0	0
1	0	0	0	0	0	1	1
0	0	0	0	0	1	0	1
0	0	0	1	0	0	0	0
0	1	0	0	0	0	0	1
1	0	0	0	1	0	0	0
0	0	0	0	0	1	0	1

Tabuľka 3.14: Stavy neurónov v čase

Výstupný reťazec $x_8 = (100110101)$ korešponduje s výsledným reťazcom vygenerovaným konečnostavovým automatom. V tabuľke 3.14 môžeme overiť aj zhodu prechodovej postupnosti. Neuróny x_2 a x_3 reprezentujú stav S_0 . Neuróny x_4 a x_5 reprezentujú stav S_1 . Neuróny x_6 a x_7 reprezentujú stav S_2 . V prípade, že jeden z dvojice neurónov mal v čase t stav rovný 1, tak daný stav bol v čase t aktívny. Z toho môžeme vyvodit' prechodovú postupnosť $(S_1, S_0, S_2, S_2, S_1, S_0, S_1, S_2, S_1)$. Nultý čas, v ktorom neurón x_4 má stav 1, ignorujeme.

Na základe vyššie popísaného postupu vyplýva, že týmto spôsobom vieme vytvorit' rekurentnú neurónovú sieť ekvivalentnú s konečnostavovým automatom a pre každý stav potrebujeme vytvorit' 2 prechodové neuróny a do výstupného neurónu smerujeme výstupy prechodových neurónov, ktorých výstup má hodnotu 1 pri ich aktivácií. Nesmieme však zabudnúť vždy na začiatku testovania neurónovej siete aktivovať prechodový neurón v čase t , ktorý vstupuje v čase $t+1$ do začiatočného stavu. V prípade, že neexistuje stav vracajúci sa do začiatočného stavu, je potrebné vytvorit' umelý neurón s počiatočným

stavom 1, ktorý vstupuje do začiatočného neurónu a aktivuje začiatočný stav pri štarte neurónovej siete.

Počet neurónov potrebných pre vytvorenie rekurentnej neurónovej siete ekvivalentnej s konečnosťavým automatom, ktorého aspoň jeden prechod má výstupnú hodnotu 1, vieme vyjadriť funkciou:

$$n = 2 * S + 2$$

kde:

- *n je počet neurónov*
- *S je počet stavov konečnosťavového automatu*

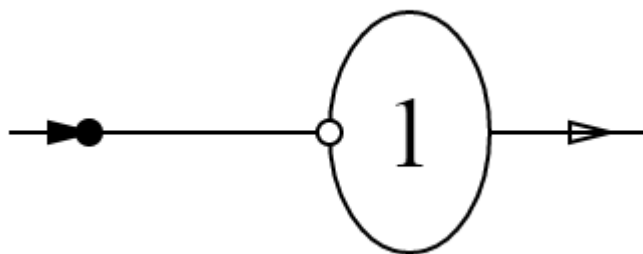
3.3 Porovnanie rôznych spôsobov prevodu konečnosťavového automatu na neurónovú sieť

V kapitole 3.2 sme popísali a overili tri rôzne spôsoby tvorby neurónovej siete ekvivalentnej ku konečnosťavému automatu. Tieto postupy sa odlišujú spôsobom, akým neurónová sieť generuje výstupný reťazec. Iný spôsob akým reprezentovať prechodovú logiku automatu neurónmi sme neobjavili a dospeli sme k rovnakému spôsobu, aký bol popísaný aj v práci Efficient Simulation of Finite Automata by Neural Nets, kde v závere uviedli: „*We saw that while some m-state machines can be built using very few neurons, some need many more neurons to be built. A naturally arising question is whether there are any interesting classes of finite automata that can be efficiently built as neural nets; namely, with a number of neurons that is very small compared with the size of the automata. Preliminary research indicates that while there are classes of finite state machines that can be efficiently simulated in this sense, all really interesting classes considered so far are too rich in structure and size. This research is continuing.*“[19] Z tohto dôvodu sa odporúča pred prevodom konečnosťavového automatu na rekurentnú neurónovú sieť zminimalizovať konečnosťavý automat, aby sa zminimalizoval počet stavov automatu a tým pádom aj počet dvojíc-trojíc prechodových neurónov reprezentujúcich stav automatu.

Všetky postupy prevodu konečnostavového automatu na rekurentnú neurónovú sieť popísané v kapitole 3.2 sa dajú použiť na N-stavový automat a tým pádom sú univerzálne bez ohľadu na počet stavov automatu.

Pri všetkých troch postupoch, ktoré sme popísali v kapitole 3.2 nesmieme zabudnúť vždy na začiatku testovania neurónovej siete aktivovať prechodový neurón v čase t , ktorý vstupuje v čase $t+1$ do začiatočného stavu. V prípade, že neexistuje stav vracajúci sa do začiatočného stavu, je potrebné vytvoriť umelý neurón s počiatočným stavom 1, ktorý vstupuje do prechodových neurónov, ktoré reprezentujú začiatočný stav automatu a tým aktivuje tento začiatočný stav pri štarte neurónovej siete.

V prípade, že by zadaný konečnostavový automat mal vo všetkých možných prechodoch výstup s hodnotou 0, v neurónovej sieti vytvorenej postupom opísaným v kapitole 3.2.3 by neexistoval prechodový neurón smerujúci do výstupného neurónu. Zároveň by v neurónovej sieti vytvorenej postupom opísaným v kapitole 3.2.2 neexistovala výstupná funkcia podľa ktorej by sa vytvorila výstupná neurónová vrstva. V takomto prípade by sme sa držali vety: „Pre ekvivalenciu neurónovej siete s konečnostavovým strojom nie je podstatné akým spôsobom sa vstupné symboly zobrazujú na výstupné symboly, t.j. akým spôsobom prebiehajú výpočty pri konštrukcii výstupného reťazca, podstatná je rovnosť výstupných reťazcov pre rovnaký vstupný reťazec pre obe zariadenia“ [2] a zostrojili by sme neurónovú sieť s jedným neurónom (obrázok 3.15), do ktorého by inhibične vstupovala vstupná hodnota a prah tohto neurónu by mal hodnotu 1. Tento neurón by bol zároveň výstupným neurónom. Takto vytvorená neurónová sieť by pri akejkoľvek binárnej postupnosti generovala výstupný reťazec tvorený iba hodnotou 0.



Obr. 3.15: Neurónová sieť s nulovým výstupom v každom prechode

Jednotlivé spôsoby prevodu sa líšia nie len spôsobom výstupu, ale aj zložitou výpočtového procesu a počtom potrebných neurónov pre vytvorenie takejto neurónovej siete.

V prvom spôsobe tvorby neurónovej siete opísanom v kapitole 3.2.1 je výpočtová zložitost' minimálna. Prechodové neuróny jednotlivých stavov majú vždy rovnaký tvar a líšia sa len počtom vstupov, ktorý závisí od toho, koľko stavov smeruje do daného stavu v konečnostavovom automate. Ďalej stačí zistiť, ktorá zo štyroch kombinácií (00, 01, 10 a 11) výstupných hodnôt nastáva v danom stave a podľa toho zvoliť výstupný neurón (obrázok 3.4, 3.5, 3.6 alebo 3.10) pre daný stav. Výstupy týchto výstupných neurónov jednotlivých stavov sa spájajú vo výstupnom neuróne s logickou operáciou OR. Postupnosť algoritmov pre implementáciu grafického prevodu pre tento spôsob je nasledovná:

Vykresli vstupný neurón

Zisti, o aký typ výstupu sa jedná (pre každý stav)

Vykresli výstupný neurón stavu (pre každý stav)

Vykresli prechodové neuróny (pre každý stav)

Vykresli výstupný neurón siete

Vykresli spoj vstupnej hodnoty do výstupného neurónu stavu (pre každý stav)

Vykresli spoj vstupnej hodnoty do prechodových neurónov (pre každý stav)

Vykresli spoj výstupného neurónu stavu do výstupného neurónu siete (pre každý stav)

Vykresli spoj prechodového neurónu stavu do prechodových neurónov stavu a výstupného neurónu stavu, do ktorého prechádza (pre každý prechod)

V druhom spôsobe tvorby neurónovej siete opísanom v kapitole 3.2.2 je výpočtová zložitost' väčšia ako v spôsobe opísanom v kapitole 3.2.1. Prechodové neuróny majú rovnakú logiku ako v predošlom spôsobe. Výstup sa ale realizuje ďaleko zložitejším spôsobom. V tomto spôsobe prevodu treba pomocou Karnaughovej mapy zistiť logickú reprezentáciu výstupnej funkcie a tú následne zminimalizovať na základe znalosti, že v čase t je aktívny vždy len jeden prechodový stav. Výpočet Karnaughovej mapy pre viac premenných je samo o sebe zložitý algoritmus a mnohé z online dostupných výpočtov pre Karnaughovu mapu sú obmedzené počtom premenných, registráciou užívateľa alebo dotáciou na beh servera, nakoľko tento algoritmus výrazne vyťažuje jadrá serverov, na

ktorých sú umiestnené. Nehovoriac o tom, že bežné algoritmy pre výpočet Karnaughovej mapy by museli byť upravené pre použitie na zistenie logickej interpretácie výstupnej funkcie, nakoľko Karnaughova mapa pre výstupnú funkciu obsahuje aj situácie, ktoré nikdy nenastanú a je len na nás, či si dané pozície obsadíme hodnotou 1 alebo 0. Toto rozhodovanie musí byť však inteligentné, aby sa docielilo zminimalizovanie výstupnej funkcie. Postupnosť algoritmov pre implementáciu grafického prevodu pre tento spôsob je nasledovná:

Vykresli vstupný neurón

Vyrátaj Karnaughovu mapu pre výstupnú funkciu

Optimalizuj Karnaughovu mapu pre minimalizovanie výstupnej funkcie

Vykresli všetky neuróny potrebné pre reprezentáciu výstupnej funkcie

Vykresli prechodové neuróny (pre každý stav)

Vykresli spoj vstupnej hodnoty do prechodových neurónov (pre každý stav)

Vykresli spoj prechodového neurónu stavu do prechodových neurónov stavu, do ktorého prechádza (pre každý prechod)

Vykresli všetky spoje potrebné pre výstupnú funkciu

V treťom spôsobe tvorby neurónovej siete opísanom v kapitole 3.2.3 je výpočtová zložitosť opäť minimálna. Prechodové neuróny majú rovnakú logiku ako v predošlých spôsoboch. Pre výstup sa použije jeden neurón s prahom 1, na ktorého vstup vstupujú exitačné výstupy prechodov, ktorých výstup v automate má hodnotu 1. Postupnosť algoritmov pre implementáciu grafického prevodu pre tento spôsob je nasledovná:

Vykresli vstupný neurón

Vykresli prechodové neuróny (pre každý stav)

Vykresli výstupný neurón siete

Vykresli spoj vstupnej hodnoty do prechodových neurónov (pre každý stav)

Vykresli spoj prechodového neurónu stavu do výstupného neurónu siete (pre každý prechod, ktorého výstup je 1)

Vykresli spoj prechodového neurónu stavu do prechodových neurónov stavu, do ktorého prechádza (pre každý prechod)

Porovnaním týchto spôsobov prevodu s ohľadom na počet neurónov potrebných pre vytvorenie rekurentnej neurónovej siete ekvivalentnej s konečnostavovým automatom

a s ohľadom na výpočtovú zložitosť jasne vyplýva, že prevod opísaný v kapitole 3.2.3 je najefektívnejší spôsob prevodu konečnostavového automatu na neurónovú sieť.

4 Implementácia

V kapitole 3.3 sme porovnali a vybrali najefektívnejší spôsob prevodu konečnostavového automatu s N stavmi na rekurentnú neurónovú sieť. Samotná implementácia tohto prevodu je popísaná v tejto kapitole.

4.1 Výber programovacieho jazyka

Na implementáciu prevodu opísaného v prílohe A a v kapitole 2 sme použili jazyk HTML, CSS a JavaScript. Vďaka tejto kombinácii sme vytvorili program, ktorý funguje na všetkých operačných systémoch, na všetkých zariadeniach a vo všetkých webových prehliadačoch. Zároveň sme nenašli žiadne obmedzenia či iné prekážky, ktoré by nás prinútili zmeniť našu predošlú voľbu.

Pre vykresľovanie sme vybrali JavaScriptovú knižnicu Raphaël (<http://raphaeljs.com/>) pre vektorové vykresľovanie. Knižnica Raphaël je chránená licenciou MIT, ktorá je súčasťou súboru raphael.js.

Používateľské prostredie prešlo mnohými zmenami oproti predošlej verzii. Jednou z najväčších zmien je responzivnosť aplikácie.

4.2 Zadávanie vstupných parametrov

Pre zadávanie vstupných parametrov sme zvolili prechodovú a výstupnú tabuľku konečnostavového automatu vďaka čomu rapídne znižujeme chybovosť, ktorú sme odhalili pri predchádzajúcom formulári, kde sa parametre automatu zadávali cez jednotlivé prechody.

Pri predošlom vzhľade formulára pre zadávanie parametrov konečnostavového automatu sa používateľ mohol pomýliť zadaním dvoch prechodov smerujúcich

z rovnakého stavu pri rovnakom vstupe do rôznych stavov. Predchádzajúci formulár vyzeral nasledovne:

Počet stavov automatu:

Počet prechodov medzi stavmi:

Zo stavu :	vstupná hodnota :	výstupná hodnota :	do stavu	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Zo stavu :	vstupná hodnota :	výstupná hodnota :	do stavu	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Zo stavu :	vstupná hodnota :	výstupná hodnota :	do stavu	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Zo stavu :	vstupná hodnota :	výstupná hodnota :	do stavu	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Pri implementácii prevodu opísaného v kapitole 3.2.3 sme pre zadávanie parametrov konečnostavového automatu zvolili prechodovú a výstupnú tabuľku. Nová verzia formuláru vyzera nasledovne:

Počet stavov automatu:

	Prechod		Výstup	
	0	1	0	1
Stav 0 :	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Stav 1 :	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Stav 2 :	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Stav 3 :	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Nová verzia formuláru má zároveň vylepšené chybové oznámenia a umožňuje používateľovi opraviť zadané údaje a znova odoslať. Podrobný návod pre zadávanie vstupných parametrov je v prílohe C.

4.3 Vykresľovanie rekurentnej neurónovej siete

Po načítaní a kontrole vstupných parametrov nasledujú dve logické kontroly:

- kontrola potreby neurónu štartujúceho sieť (v prípade, že neexistuje prechod smerujúci do začiatočného stavu) – funkcia `zistiumelystav(states)`
- kontrola, či existuje aspoň jeden výstup s hodnotou 1 (v opačnom prípade stačí vykresliť neurónovú sieť na obrázku 3.15) – funkcia `nulovy_vystup(states)`

```
var pomocna = zistiumelystav(states);

//zisti, ci máme nejaký výstup s hodnotou 1
var pomocna2 = nulovy_vystup(states);
if(pomocna2 == 0) { //vykresľujeme neuronovú sieť s nulovým
vystupom vo všetkých prípadoch
    content += ("Automat má vo všetkých prípadoch nulový výstup.
Vykresľujem zjednodušenú sieť.<br/>");
    vykresliNulovuSiet();
}
else {
    vykresliSiet(states);

    if(pomocna == 0){ //vykresľujeme pomocný neuron pre potreby
aktivacie zaciatočného neuronu
        content += ("Žiadny prechod nesmeruje do začiatočného
stavu. Vykresľujem umelý stav pre štart siete (viď dokumentácia).<br/>");
        vykresliUmelýStav();
    }
}
```

Ak sú všetky údaje v poriadku a existuje aspoň jeden výstup s hodnotou 1, môžeme začať s vykresľovaním rekurentnej neurónovej siete. O vykreslenie sa stará funkcia `vykresliSiet(states)`, ktorá dostáva na vstup parameter `states`, ktorý reprezentuje počet stavov automatu.

```
function vykresliSiet(states) {

    paper = Raphael("content5", 680, states*200); //vytvor priestor pre
kreslenie

    vykresliVstup();
    vykresliVystup(states);
    vykresliPrechodoveNeurony(states);
    vykresliPrechody(states);
}
```

Funkcia `vykresliVstup()` vykreslí začiatočnú čiaru, šípku a guľičku, ktorá reprezentuje vstup vstupných hodnôt do neurónovej siete.

Funkcia *vykresliVystup(states)* vykreslí výstupný neurón s prahom 1, čiaru z tohto neurónu a šípku. Zároveň táto funkcia skontroluje výstupnú funkciu a vykreslí exitačné vstupy z tých prechodov, ktoré majú pri aktivácii výstup 1.

Funkcia *vykresliPrechodoveNeurony(states)* vykreslí pre každý stav automatu dva neuróny. Jeden s inhibičným vstupom vstupnej hodnoty a prahom 1 a druhý s exitačným vstupom vstupnej hodnoty a prahom 2.

Funkcia *vykresliPrechody(states)* vykreslí výstupy prechodových neurónov smerujúcich na vstup do dvojíc neurónov reprezentujúcich nasledujúci stav.

```
function vykresliPrechody(states) {
    var i, x, pomocna;
    var vstupy = new Array(states); //kolko prechodov vstupuje do stavu
    var aktualne = new Array(states); //kolko z tychto prechodov uz mam
    vykreslenych

    for (i=0; i < states; i++) { //vynulujeme polia
        vstupy[i]=0;
        aktualne[i]=0;
    }
    for (i=0; i < states; i++) { //naplnime pole vstupy - kolko
    prechodov do stavu prechadza
        pomocna=pole[i][0];
        vstupy[pomocna]++;
        pomocna=pole[i][1];
        vstupy[pomocna]++;
    }
    vykresliExitVstupy(states, vstupy);
    vykresliPrechodoveVstupy(states, aktualne);
}
```

Aby každý výstup prechodu mal v nasledujúcom stave reprezentovanom dvomi neurónmi vlastný exitačný vstup, použili sme dôkladnejšie vykresľovanie 3 pomocné premenné: pole *vstupy*, pole *aktualne* a parameter *pocetvykreslenych*. Polia *vstupy* a *aktualne* majú veľkosť *states*. Do poľa *vstupy* sa vloží údaj, koľko prechodov prechádza do daného stavu. Následne funkcia *vykresliExiVstupy(states, vstupy)* vykreslí pre každý prechod exitačný vstup pre dvojicu neurónov.

```
function vykresliExitVstupy(states, vstupy) {

    var i, x;
    for (i=0; i < states; i++) {
        for (x=0; x < vstupy[i]; x++) {
            var circle1 = paper.circle(280, 37+i*160+x*8, 3);
            //vykresli kruh - exitacny vstup, neuron 1
        }
    }
}
```

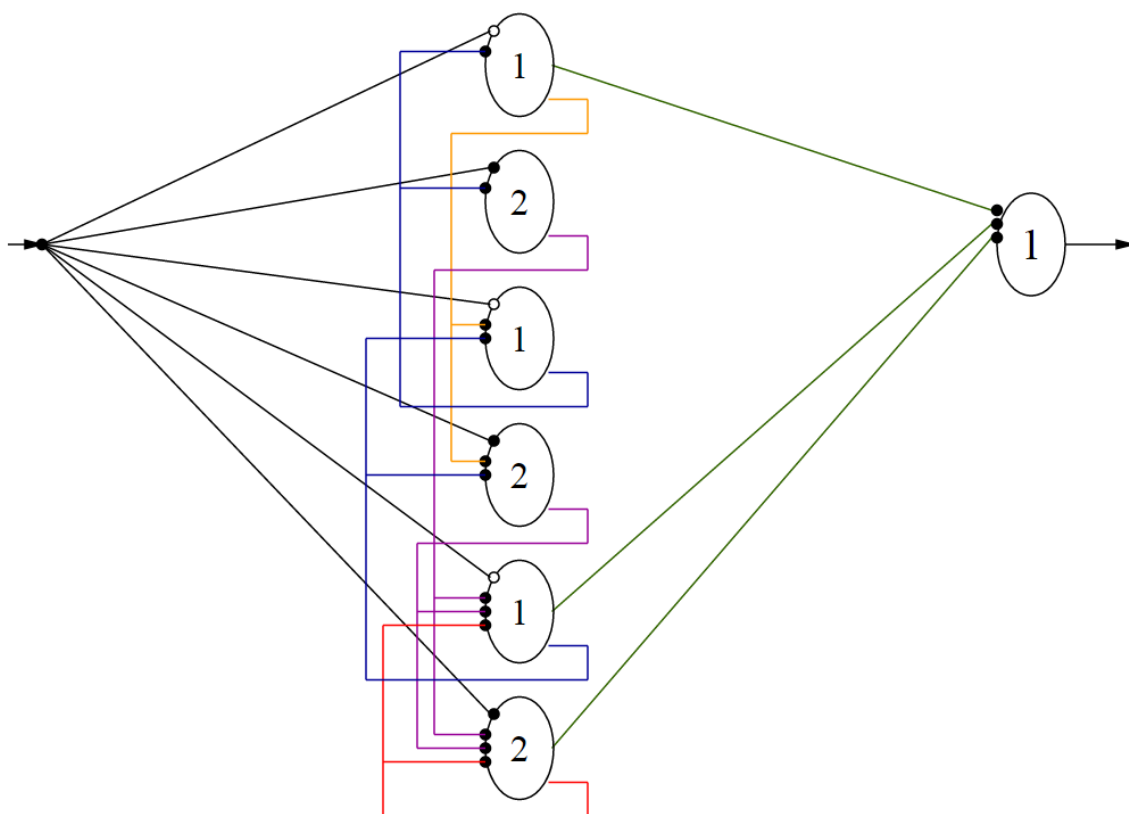
```

        circle1.attr("fill", "#000000");
        circle1.attr("stroke", "#000000");
        var circle2 = paper.circle(280, 117+i*160+x*8, 3);
//vykresli kruh - exitacny vstup pre neuron 2
        circle2.attr("fill", "#000000");
        circle2.attr("stroke", "#000000");
    }
}
}

```

Následne funkcia *vykresliPrechody(states)* spustí funkciu *vykresliPrechodoveVstupy(states, aktualne)*. V poli *aktualne* si zaznačujeme údaj, koľko z týchto prechodov do daného stavu sme už vykreslili, aby sme presne vedeli, do ktorého exitačného vstupu má daný prechod vstúpiť. V parametri *pocetvykreslenych*, ktorú používame vo funkcií *vykresliPrechodoveVstupy(states, aktualne)* si ukladáme hodnotu, koľko prechodov sme už vykreslili. Na základe tohto parametru vykresľujeme prechod vo väčšej vzdialenosti a zabraňujeme tým prekrývaniu sa jednotlivých prechodov.

Po zadaní parametrov konečnostavového automatu z kapitoly 3.1 program vygeneruje rekurentnú neurónovú sieť zobrazenú na obrázku 4.1. Z obrázku vyplýva, že jednotlivé prechody sú oddelené a každý prechod má svoj vlastný exitačný vstup do neurónu.



Obr. 4.1: Rekurentná neurónová sieť vygenerovaná programom

5 Záver

„Pre ekvivalenciu neurónovej siete s konečnosťavým strojom nie je podstatné akým spôsobom sa vstupné symboly zobrazujú na výstupné symboly, t.j. akým spôsobom prebiehajú výpočty pri konštrukcii výstupného reťazca, podstatná je rovnosť výstupných reťazcov pre rovnaký vstupný reťazec pre obe zariadenia.“[2]

V kapitole 2 sme popísali spôsob prevodu konečnosťavového automatu na rekurentnú neurónovú sieť podľa schémy na obrázku 2.2, ktorú popísal prof. Ing. Vladimír Kvasnička, DrSc. vo svojej knihe [2]. Tento spôsob prevodu ale nie je postačujúci pre prevod konečnosťavového automatu s viac ako dvoma stavmi. Cieľom tejto práce preto bolo objaviť a graficky implementovať prevod konečnosťavového automatu s N stavmi na neurónovú sieť.

V kapitole 3 sa nám podarilo opísať až tri rôzne spôsoby tvorby rekurentnej neurónovej siete ekvivalentnej zadanému konečnosťavovému automatu s N stavmi. Na príklade sme ukázali ako vytvoriť neurónovú sieť pomocou všetkých troch spôsobov. Táto sieť je ekvivalentná k automatu pri akejkol'vek postupnosti vstupných hodnôt. Zložitosti týchto troch spôsobov sme porovnali v kapitole 3.3. V kapitole 4 sme zbežne popísali postup implementácie najefektívnejšieho z troch spôsobov prevodu a popísali sme významné funkcie praktickej časti tejto práce.

Výsledkom tejto práce je teoretický a zároveň aj praktický prevod konečnosťavového automatu s N stavmi na rekurentnú neurónovú sieť. Aplikačná časť je dostupná aj online na webovej stránke www.xkatka.sk/dp/. Aplikačná časť bola vytvorená tak, aby bola ľahko použiteľná a to bez potreby akejkol'vek inštalácie či nutnosti výberu vhodného operačného systému alebo zariadenia.

Spojením výsledku tejto práce s predošlým výskumom (príloha A) sme zavřšili grafický prevod konečnosťavového automatu na rekurentnú neurónovú sieť a naopak. Výsledok tejto práce môže byť používaný pedagogickými pracovníkmi pri výučbe neurónových sietí, konečnosťavových zariadení, kognitívnej vedy, atď.

V prípade ďalšieho vývoja programu by bolo vhodné implementovať rôzne spôsoby zadávania vstupných dát, ich možný export a opätovný import, ako aj import vstupných dát z iných programov či export výsledku do iných programov. Pre plnohodnotné využitie znalostí o ekvivalenciách konečnostavového automatu a neurónovej siete v ekonómii je potrebné vyškoliť ekonomických pracovníkov pre prácu s konečnostavovým automatom a implementovať export výsledku tohto prevodu do programu, ktorý ekonomickí pracovníci používajú v spojitosti s neurónovými sieťami pri svojej práci.

Literatúra

- [1] KVASNIČKA, V. a kol. 1997. *Úvod do teórie neurónových sietí*. Bratislava: Vydavateľstvo IRIS, 1997. 262 s. ISBN 80-88778-30-1.
- [2] KVASNIČKA, V. - POSPÍCHAL, J. 2006. *Matematická logika*. Bratislava: STU vydavateľstvo, 2006. 389 s. ISBN 80-22724-49-1.
- [3] MAKULA, Matej. 2009. *Sila a obmedzenia rekurentných neurónových sietí pri spracovaní postupností symbolov*: dizertačná práca. Bratislava: FIIT STU, 2009. 76 s.
- [4] VANÍČEK, Jiří a kol. 2007. *Teoretické základy informatiky*. 1.vyd. Bratislava: Alfa Publishing, 2007. 431 s. ISBN 80-903962-4-1
- [5] BAYER, J. - HANZÁLEK, Z. - ŠUSTA, R. 2000. *Logické systémy pro řízení*. Praha: Vydavateľstvo ČVUT, Fakulta elektrotechnická. 2000. 269 s. ISBN 80-01-02147-5.
- [6] ČERNĀNSKÝ, M. 2008. *Logické neuróny a neurónové siete* (podľa McCullocha a Pittsa) [online]. Bratislava: FIIT STU. 2008. [študované 3.4.2015] Dostupné na internete:
<http://www2.fiit.stuba.sk/~cernans/nn/nn_texts/neuronove_siete_priesvitky_01.pdf>
- [7] CHUDÁ, D. a kol. 2006. *Teoretické základy informatiky* [online]. Bratislava: FIIT STU. 2006. 54 s. Dostupné na internete:
<<http://student.fiit.stuba.sk/~zuber06/TZI/Ucebny%20mat%20kap%201%20teoria%20mnozin.pdf>>
- [8] TREBATICKÝ, Peter. *Trénovanie rekurentných neurónových sietí technikou rozšíreného Kalmanovho filtra* [online]. [cit. 2015.4.3] Dostupné na internete:
<<http://acm.vsb.cz/is/2004/finale/prispevky/trebaticky.pdf>>
- [9] PAVLECH, Michal. 2009. *Neurónové siete v C#*: diplomová práca. Zlín: UTB, Fakulta aplikované informatiky. 2009. 94 s.
- [10] BAHNA, R. 1998. *Využitie neurónových sietí na riešenie logických problémov*: diplomová práca. Bratislava: FEI STU. 1998.
- [11] ŠŤASTNÝ, J. - POKORNÝ, M. - MOTYČKA, A. 2007. *Aplikace umělé inteligence v ekonomické oblasti* [online]. Brno: Mendelova zemědělská a

- lesnická univerzita. 2007. 129-135 s. Dostupné na internete:
<http://www.ki.fpv.ukf.sk/projekty/kega_3_4029_06/iski2007/papers/Stastny%20Pokorny%20Motycka.pdf>
- [12] FRANCISCO GARCIA FERNANDEZ - IGNACIO SORET LOS SANTOS - JAVIER LOPEZ MARTINEZ - SANTIAGO IZQUIERDO IZQUIERDO - FRANCISCO LLAMAZARES REDONDO. 2013. *Use of Artificial Neural Networks to Predict The Business Success or Failure of Start-Up Firms, Artificial Neural Networks - Architectures and Applications*. ISBN: 978-953-51-0935-8, InTech, DOI: 10.5772/51381. Dostupné na internete:
<<http://www.intechopen.com/books/artificial-neural-networks-architectures-and-applications/use-of-artificial-neural-networks-to-predict-the-business-success-or-failure-of-start-up-firms>>
- [13] KATE A. SMITH - JATINDER N.D. GUPTA. 2000. *Neural networks in business: techniques and applications for the operations researcher* [online]. 2000. 1024-1044 s. [cit. 2015.4.12] Dostupné na internete:
<http://sedok.narod.ru/s_files/poland/112.pdf>
- [14] Mohsen Nazari - Mojtaba Alidadi. 2013. *Measuring Credit Risk of Bank Customers Using Artificial Neural Network* [online]. Teherán: University of Tehran. 2013. 17-27 s. [cit. 2015.4.29] Dostupné na internete:
<<http://macrothink.org/journal/index.php/jmr/article/viewFile/2899/2621>>
- [15] TIÑO P. - BILL G. HORNE - GILES C. LEE - COLLINGWOOD PETE C. 1998. *Finite State Machines and Recurrent Neural Networks - Automata and Dynamical Systems Approaches*. Washington: University of Maryland. 1998. 49 s.
- [16] CLEEREMANS A. - D. SERVAN-SCHREIBER - J.L. MCCLELLAND. 1989. *Finite state automata and simple recurrent networks. Neural Computation*. Massachusetts: Institute of Technology. 1989. 372-381 s. ISSN: 0899-7667. DOI: 10.1162/neco.1989.1.3.372.
- [17] B.G. HORNE - D.R. HUSH. 1996. *Bounds on the complexity of recurrent neural network implementations of finite state machines*. Oxford: Elsevier Science Ltd. 1996. 359-366 s. DOI: 10.1016/0893-6080(95)00095-X.
- [18] JOHN F. KOLEN. 1994. *Fool's Gold: Extracting Finite State Machines From Recurrent Network Dynamics*. Columbus: The Ohio State University. 1994. 501-508 s.

- [19] NOGA ALON - A. K. DEWDNEY - TEUNIS J. OTT. 1991. *Efficient Simulation of Finite Automata by Neural Nets*. New York: ACM. 1991. 495-514 s. DOI: 10.1145/103516.103523.

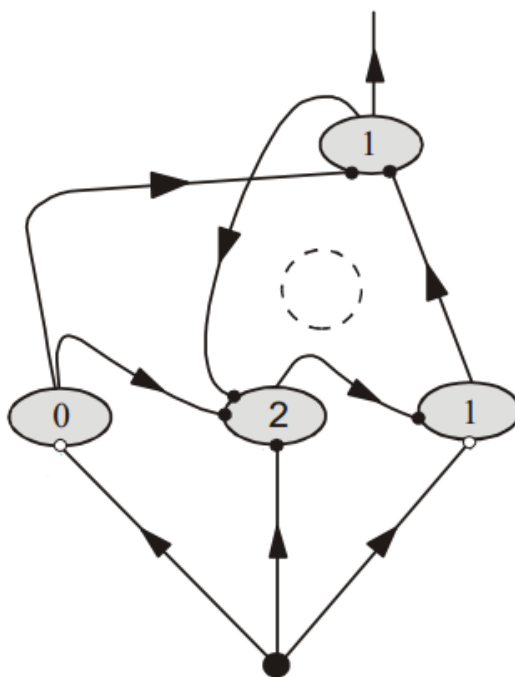
Príloha A Prevod neurónovej siete na konečnosťavový automat

Na prevod neurónovej siete potrebujeme poznať parametre siete a vstupné hodnoty. Parametre neurónovej siete môžu byť rôzne. Sieť vieme definovať funkciou, ktorú reprezentuje, taktiež funkciami, ktoré reprezentujú jednotlivé neuróny siete alebo aj jednotlivými prechodmi medzi neurónmi.

A.1 Zadefinovanie neurónovej siete

Pri definícii neurónovej siete pomocou funkcie siete môžu byť výsledky rozdielne, nakoľko rozklad zloženej funkcie na lineárne separovateľné funkcie môže mať viac ako jedno riešenie. Z tohto dôvodu ako aj z dôvodu jednoduchosti zadávania parametrov siete je optimálnejšie definovať parametre siete cez jednotlivé prechody medzi neurónmi.

Prvý krok prevodu neurónovej siete na konečnosťavový automat spočíva v definovaní stavov neurónov v čase t . Dĺžka času závisí od počtu vstupných hodnôt. Predpokladáme, že stavy (resp. aktivity) neurónov v čase $t=1$ sú nulové. Výstup neurónovej siete v čase $t=1$ je prázdny znak.



Obr. A.1: Rekurentná neurónová sieť

Aktivity neurónov neurónovej siete na obr. A.1 pri vstupných hodnotách $x_1 = (110110101010)$ sú zobrazené v tabuľke A.1

	x_1	x_2	x_3	x_4	x_5
1	1	0	0	0	-
2	1	0	0	0	0
3	0	0	0	0	0
4	1	1	0	0	0
5	1	0	1	0	1
6	0	0	1	0	0
7	1	1	0	1	0
8	0	0	1	0	1
9	1	1	0	1	0
10	0	0	1	0	1

Tabuľka A.1: Aktivita neurónov v čase – vstup a výstup siete

Stĺpec x_1 reprezentuje vstupné hodnoty a stĺpec x_5 v tabuľke A.1 reprezentuje výstupné hodnoty rekurentnej neurónovej siete z obrázka A.1. Pre výpočet aktivity neurónov v čase používame „pamäť“ sietí. „Rekurentné neurónové siete sú na rozdiel od klasických dopredných neurónových sietí schopné spracovávať vstupy s časopriestorovou štruktúrou.“ [8] Napríklad pri výpočte stavu neurónu x_4 v čase t potrebujeme poznať stav x_1 a x_3 v čase $t-1$. Pre zistenie stavu neurónu x_3 v čase $t-1$ musíme zase poznať stav neurónu x_1 , x_2 a x_5 v čase $t-2$. Túto vlastnosť špecifickú pre rekurentné neurónové siete nazývame „pamäť“ sietí.

Aktivity neurónov zo siete na obrázku A.1:

$$x_1^{(t)} = \text{externý vstup}$$

$$x_2^{(t)} = s(-x_1^{(t-1)} - 0)$$

$$x_3^{(t)} = s(x_1^{(t-1)} + x_2^{(t-1)} + x_5^{(t-1)} - 2)$$

$$x_4^{(t)} = s(-x_1^{(t-1)} + x_3^{(t-1)} - 1)$$

$$x_5^{(t)} = s(x_2^{(t-1)} + x_4^{(t-1)} - 1)$$

A.2 Tvorba konečnostavového automatu

Konečnostavový automat, reprezentujúci neurónovú sieť danú parametrami siete a vstupnými hodnotami, bude obsahovať stavy, ktoré predstavujú skryté neuróny siete. Stavy konečnostavového automatu označujeme hodnotou, ktorá reprezentuje hodnoty aktivít všetkých skrytých neurónov. Konečnostavový automat, ktorý reprezentuje neurónovú sieť zobrazenú na obrázku A.1, začína stavom S_{000} , nakoľko aktivity troch skrytých neurónov v čase $t=1$ sú nulové.

V prípade rekurentnej neurónovej siete z obrázka A.1 s aktivitami neurónov definovanými v tabuľke A.2 vyplýva, že na reprezentáciu siete automatom potrebujeme 4 stavy (S_{000} , S_{100} , S_{010} a S_{101}).

	x_1	x_2	x_3	x_4	x_5
1	1	0	0	0	-
2	1	0	0	0	0
3	0	0	0	0	0
4	1	1	0	0	0
5	1	0	1	0	1
6	0	0	1	0	0
7	1	1	0	1	0
8	0	0	1	0	1
9	1	1	0	1	0
10	0	0	1	0	1

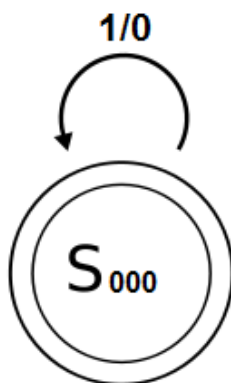
Tabuľka A.2: Aktivita neurónov v čase – stavy skrytých neurónov

Prechody medzi stavmi konečnostavového automatu sú definované východiskovým stavom, vstupnou hodnotou, výstupnou hodnotou a výstupným stavom. Tieto prechody môžeme jednoducho vyčítať z tabuľky aktivity neurónov v čase. Vstupný symbol, teda stav neurónu x_1 v čase t , je vstupnou hodnotou automatu. Výstupný symbol, teda stav neurónu x_5 v čase $t+1$, je výstupnou hodnotou automatu. Stavy skrytých neurónov v čase t reprezentujú východiskový stav automatu. Stavy skrytých neurónov v čase $t+1$ reprezentujú výstupný stav automatu.

	x_1	x_2	x_3	x_4	x_5
1	1	0	0	0	-
2	1	0	0	0	0

Tabuľka A.3: Aktivita neurónov v čase $t=1$ a $t=2$ – prechod konečnostavového automatu

Prvý prechod konečnostavového automatu na obrázku A.2, reprezentujúceho rekurentnú neurónovú sieť z obrázka A.1, vychádza zo začiatočného stavu S_{000} so vstupnou hodnotou 1, výstupný stav je opäť S_{000} s výstupnou hodnotou 0. Prvý a druhý prechod je totožný.

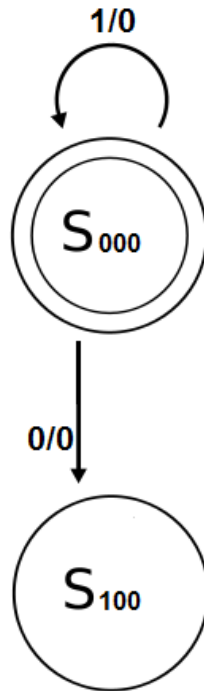


Obr. A.2: Prvý prechod konečnostavového automatu

	x_1	x_2	x_3	x_4	x_5
3	0	0	0	0	0
4	1	1	0	0	0

Tabuľka A.4: Aktivita neurónov v čase $t=3$ a $t=4$

Tretí prechod konečnostavového automatu na obrázku A.3, reprezentujúceho rekurentnú neurónovú sieť z obrázka A.1, vychádza zo stavu S_{000} so vstupnou hodnotou 0, výstupný stav je S_{100} s výstupnou hodnotou 0 podľa tabuľky A.4.

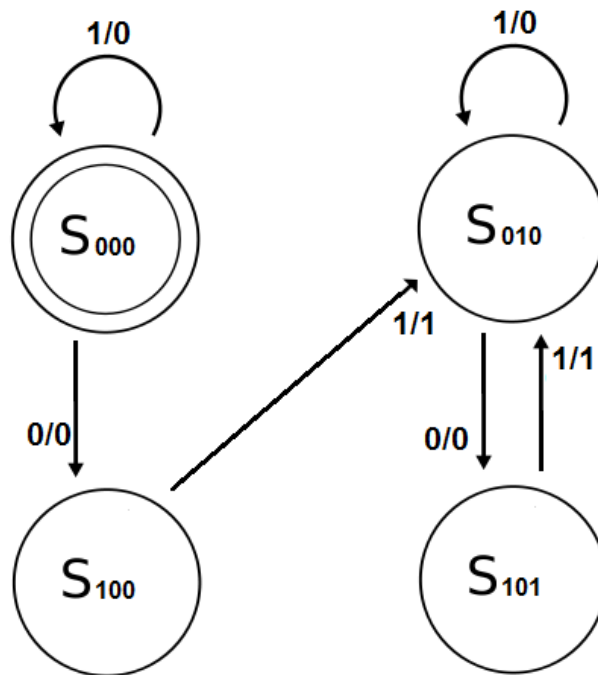


Obr. A.3: Tretí prechod konečnosťavového automatu

Opakovaním tohto kroku dostaneme konečnosťavový automat (obrázok A.4), ktorý reprezentuje rekurentnú neurónovú sieť z obrázka A.1.

Takýmto postupom získame konečnosťavový automat, ktorý reprezentuje neurónovú sieť pri zadaných vstupných hodnotách. To ale znamená, že výsledný konečnosťavový automat nemusí byť postačujúci v prípade, že dôjde k zmene vstupných hodnôt neurónovej siete, pričom nedôjde k zmene parametrov siete.

Pre vykreslenie konečnosťavového automatu, ktorý bude reprezentovať neurónovú sieť bez ohľadu na vstupný reťazec, treba 2^{n_H} stavov, kde n_H je počet skrytých neurónov siete. Stavý budú reprezentovať všetky možnosti binárnych vektorov.



Obr. A.4: Konečnostavový automat reprezentujúci neurónovú sieť

Prechody medzi stavmi v neurónovej sieti bez ohľadu na vstupné hodnoty zistíme testovaním každej hodnoty z množiny vstupných hodnôt na každom zo stavov konečnostavového automatu, resp. na každej kombinácii stavov skrytých neurónov. V prípade rekurentnej neurónovej siete nesmieme ale zabudnúť na množinu výstupov, ktoré vstupujú späť do siete. V rekurentnej sieti z obrázka A.1 testujeme 32 možností, teda $4 \cdot 8$, kde 4 je počet možných vstupov v kombinácii s výstupmi (00, 01, 10 alebo 11) a 8 je počet možných stavov (2^{n_H} , kde n_H je počet skrytých neurónov siete).

Z obrázka si zvolíme napríklad prechod zo stavu S_{000} pri vstupe 0. Výstupný stav a výstupnú hodnotu zistíme pomocou vzorcov pre výpočet aktivity neurónov v čase (viď kapitola A.1). Pri výpočte stavu neurónu x_3 , ale potrebujeme vedieť aj výstup, resp. stav neurónu x_5 v čase $t-1$.

$$x_3^{(t)} = s(x_1^{(t-1)} + x_2^{(t-1)} + x_5^{(t-1)} - 2)$$

vstup	stav			výstup
x_1	x_2	x_3	x_4	x_5
0	0	0	0	0
	1	0	0	0

Tabuľka A.5: Zisťovanie prechodu zo stavu S_{000} pri vstupe 0 a výstupe 0

vstup	stav			výstup
x_1	x_2	x_3	x_4	x_5
0	0	0	0	1
	1	0	0	0

Tabuľka A.6: Zisťovanie prechodu zo stavu S_{000} pri vstupe 0 a výstupe 1

Z výpočtov v tabuľke A.5 a A.6 vychádza, že nezáleží na predošlom výstupe, zo stavu S_{000} pri vstupnej hodnote 0 nastáva prechod do stavu S_{100} s výstupnou hodnotou 0.

Tento výpočtový krok opakujeme na stave S_{100} pri vstupe 0 v tabuľke A.7 a A.8.

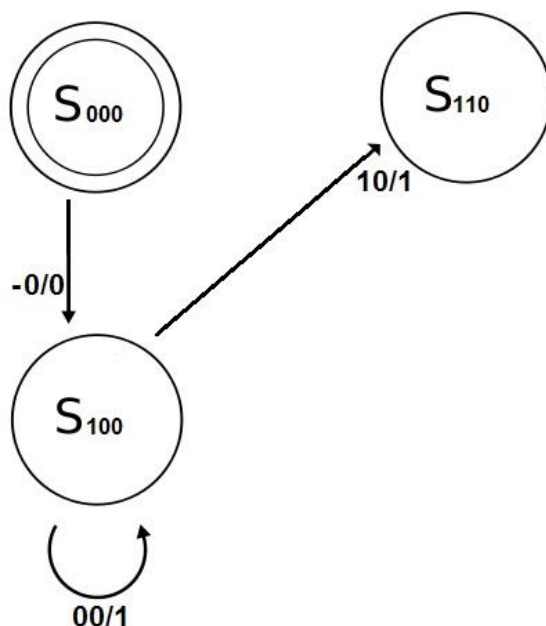
vstup	stav			výstup
x_1	x_2	x_3	x_4	x_5
0	1	0	0	0
	1	0	0	1

Tabuľka A.7: Zisťovanie prechodu zo stavu S_{100} pri vstupe 0 a výstupe 0

vstup	stav			výstup
x_1	x_2	x_3	x_4	x_5
0	1	0	0	1
	1	1	0	1

Tabuľka A.8: Zisťovanie prechodu zo stavu S_{100} pri vstupe 0 a výstupe 1

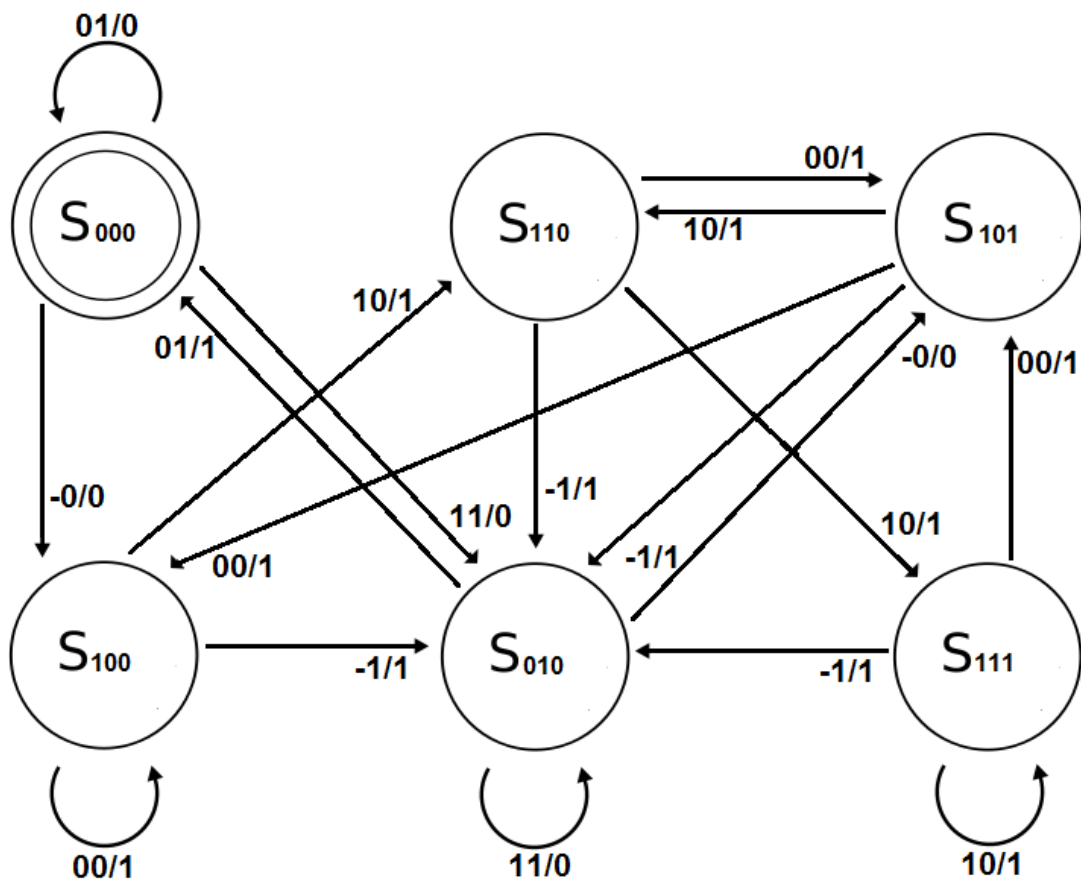
Zistili sme, že zo stavu S_{100} pri vstupe 0 s výstupom v čase $t-1$ 0 nastáva prechod v konečnostavovom automате do stavu S_{100} s výstupom 1. Zároveň sme zistili, že zo stavu S_{100} pri vstupe 0 s výstupom v čase $t-1$ 1 nastáva prechod do stavu S_{110} s výstupom 1. Teda pri tom istom vstupe nastáva zo stavu S_{100} prechod s výstupom 1 do dvoch rôznych stavov S_{100} a S_{110} v závislosti od výstupu v čase $t-1$. Túto situáciu vyriešime pridaním výstupu k vstupnej hodnote.



Obr. A.5: Konečnostavový automat s prechodmi zo stavu S_{100} pri vstupe 0 s rôznymi výstupmi v čase $t-1$

Rozšírime množinu vstupných hodnôt konečnostavového automatu. Výstup konečnostavového automatu pôjde na vstup automatu. Vstupná hodnota bude zložená z predošlého výstupu a aktuálneho vstupu. Aplikujeme túto novú skutočnosť pri vykresľovaní neurónovej siete automatom na obrázku A.5. Na obrázku A.5 sú dva prechody zo stavu S_{000} do stavu S_{100} . Jeden prechod pri výstupe v čase $t-1$ 1 a vstupnej hodnote 0, druhý pri výstupe v čase $t-1$ 0 a vstupnej hodnote 0. V rámci šetrenia priestoru pri vykresľovaní sú tieto prechody spojené do jedného, v ktorom nezáleží na výstupnej hodnote v čase $t-1$, nakoľko pri oboch možnostiach je rovnaká výstupná hodnota a výstupný stav. Výstupný stav v čase $t-1$ je v takomto prípade v množine vstupných hodnôt označovaný „-“.

Týmto postupom získame konečnosťový automat, ktorý reprezentuje neurónovú sieť bez ohľadu na vstupné hodnoty (pozri obrázok A.6).



Obr. A.6: Konečnosťový automat reprezentujúci neurónovú sieť pri akýchkoľvek vstupoch

Na obrázku A.6 je zobrazení konečnosťový automat reprezentujúci neurónovú sieť z obrázka A.1 pri všetkých možných vstupoch. Na obrázku vidíme 6 stavov, pričom na vykreslenie automatu treba 2^{n_H} stavov, kde n_H je počet skrytých neurónov siete (v našom prípade 3 skryté neuróny, čiže 8 stavov). Do zvyšných dvoch stavov S_{001} a S_{011} ale konečnosťový automat nikdy nevkročí. Túto skutočnosť si môžeme aj overiť:

Zadefinujeme si stav, ktorý chceme získať (tabuľka A.9). Teda výsledný stav S_{001} .

vstup	stav			výstup
x_1	x_2	x_3	x_4	x_5
	0	0	1	

Tabuľka A.9: Krok 1 – overenie nedosiahnuteľného stavu S_{001}

Zo vzorcov aktivity neurónov v čase vieme, že:

$$x_2^{(t)} = s(-x_1^{(t-1)} - 0)$$

$$x_3^{(t)} = s(x_1^{(t-1)} + x_2^{(t-1)} + x_5^{(t-1)} - 2)$$

$$x_4^{(t)} = s(-x_1^{(t-1)} + x_3^{(t-1)} - 1)$$

$$x_5^{(t)} = s(x_2^{(t-1)} + x_4^{(t-1)} - 1)$$

Na to, aby $x_2^{(t)}$ bolo rovné 0, musí byť hodnota $x_1^{(t-1)}$ rovná 1. Hodnota $x_3^{(t)}$ má byť rovná taktiež 0 a nakoľko už poznáme hodnotu $x_1^{(t-1)}$, zo vzorca pre výpočet aktivity neurónu x_3 v čase t vyplýva, že hodnoty $x_2^{(t-1)}$ a $x_5^{(t-1)}$ sú rovné 0. Ak hodnota $x_4^{(t)}$ má byť 1 a vieme, že hodnota $x_1^{(t-1)}$ je tiež 1, zo vzorca pre výpočet aktivity neurónu x_4 v čase t vyplýva, že hodnota $x_3^{(t-1)}$ musí byť väčšia alebo rovná 2. Táto skutočnosť ale pri binárnych stavoch neurónov nemôže nastať. Preto neexistuje situácia prechodu do stavu S_{001} v konečnostavovom automate. Rovnako postupujeme aj pri overovaní stavu S_{011} . Opäť zistíme, že hodnota $x_1^{(t-1)}$ musí byť 1. Pri zisťovaní hodnoty $x_3^{(t-1)}$ pomocou vzorca pre výpočet aktivity neurónu x_4 v čase t získame ten istý problém ako v predošlom prípade, teda hodnota $x_3^{(t-1)}$ musí byť väčšia alebo rovná 2.

vstup	stav			výstup
x_1	x_2	x_3	x_4	x_5
1	0	≥ 2		0
	0	0	1	

Tabuľka A.10: Krok 2 – overenie nedosiahnuteľného stavu S_{001}

Týmto postupom sme dokázali, že stavy S_{001} a S_{011} v konečnostavovom automate, reprezentujúcom neurónovú sieť z obrázka A.1, nemusia byť zakreslené. Tieto stavy nikdy nenastanú.

A.3 Zhodnotenie

V prílohe A.2 je podrobný popis tvorby konečnostavového automatu, ktorý je odpoveďou na neurónovú sieť so zadaným vstupným reťazcom, ako aj tvorba konečnostavového automatu, ktorý reprezentuje neurónovú sieť pri akomkoľvek vstupnom reťazci. Tento postup bol v odbornej literatúre iba zbežne opísaný, konkrétne kroky tohto prevodu sú výsledkom teoretickej časti práce na bakalárskom projekte.

V praktickej časti práce dochádza k vykresľovaniu konečnostavového automatu, ktorý reprezentuje neurónovú sieť so zadaným vstupných reťazcom. Ako je v prílohe A.2 preukázané, vykresľovanie konečnostavového automatu, ktorý reprezentuje neurónovú sieť pri akomkoľvek vstupnom reťazci, je priestorovo náročné a môže sa stať neprehľadným.

Príloha B Inštalačná príručka

Vzhľadom na výber programovacieho jazyka, program možno spustiť v internetových prehliadačoch Firefox 3.0 a vyššie, Safari 3.0 a vyššie, Chrome 5.0 a vyššie, Opera 9.5 a vyššie alebo Internet Explorer 6.0 a vyššie. Program pracuje bez problémov na všetkých operačných systémoch aj na smart telefónoch.

Program sa spúšťa otvorením súboru „index.html“ v ľubovoľnom internetovom prehliadači. V prípade, že dôjde k presunu programu na iné diskové pole, je potrebné skopírovať aj ostatné súbory a priečinky nachádzajúce sa v rovnakom priečinku ako „index.html“.

Program je dostupný aj online na webovej stránke www.xkatka.sk/dp/

Príloha C Používateľská príručka

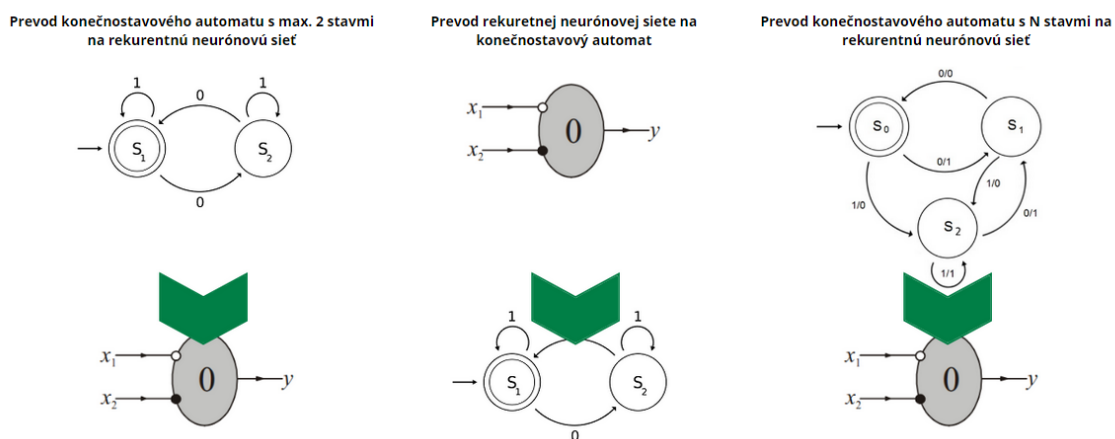
Program sa spúšťa dvoma spôsobmi:

- Online na stránke www.xkatka.sk/dp/
- Otvorením súboru `index.html` vo webovom prehliadači

Na úvodnej stránke (obrázok C.1) kliknete na tretí obrázok „Prevod konečnosťového automatu s N stavmi na rekurentnú neurónovú sieť“.

Štúdium vzťahu medzi rekurentnou neurónovou sieťou s logickými neurónmi a konečnosťovým automatom

Slovenská technická univerzita - Fakulta informatiky a informačných technológií
Ekonomická univerzita - Fakulta hospodárskej informatiky



Obr. C.1: Úvodná stránka programu

Zobrazí sa vám stránka (obrázok C.2) s formulárom pre zadanie počtu stavov automatu.

Prevod konečnosťového automatu s N stavmi na rekurentnú neurónovú sieť

Počet stavov automatu:

Vykresli

Obr. C.2: Formulár pre zadanie počtu stavov automatu

Po zadání počtu stavov automatu sa vám zobrazí ďalší formulár (obrázok C.3) pre zadanie prechodovej a výstupnej funkcie.

Počet stavov automatu:

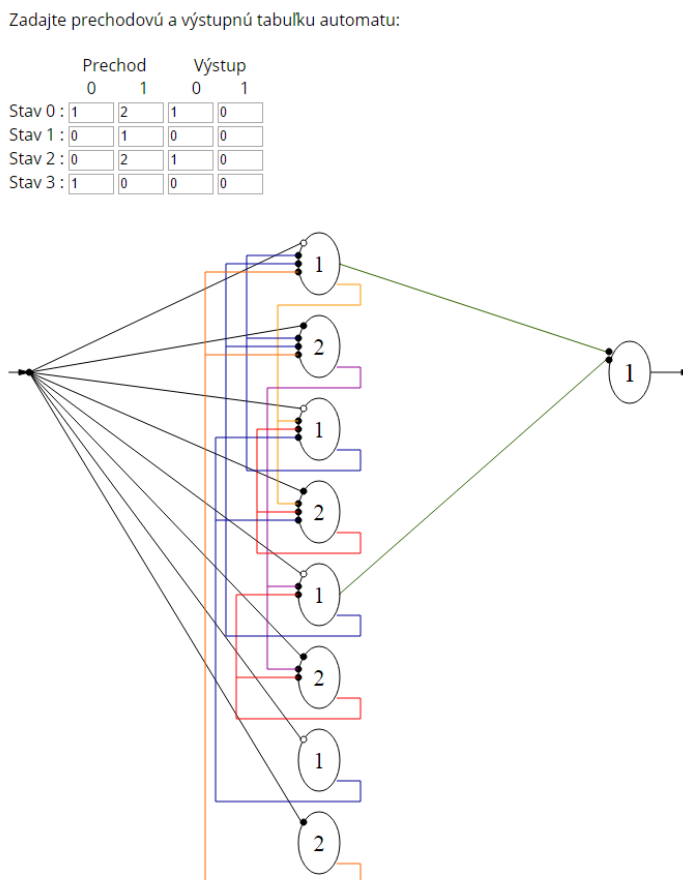
Stavy číslojte od 0, pričom nultý stav je zároveň začiatočný

Zadajte prechodovú a výstupnú tabuľku automatu:

	Prechod		Výstup	
	0	1	0	1
Stav 0 :	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Stav 1 :	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Stav 2 :	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Stav 3 :	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Obr. C.3: Formulár pre zadanie prechodovej a výstupnej funkcie

Po zadání prechodovej a výstupnej funkcie kliknite na tlačidlo „Vykresli“ a vykreslí sa vám rekurentná neurónová sieť (príklad na obrázku C.4) ekvivalentná so zadáním konečnostavovým automatom.



Obr. C.4: Príklad výstupu programu

Program akceptuje iba výstupné hodnoty 1 alebo 0. V prípade, že do tabuľky vložíte iný údaj, program vás upozorní (obrázok C.5) na nesprávne údaje a máte možnosti ich opraviť a spustiť vykresľovanie odznova.

Zadajte prechodovú a výstupnú tabuľku automatu:

	Prechod		Výstup	
	0	1	0	1
Stav 0 :	<input type="text" value="1"/>	<input type="text" value="0"/>	<input type="text" value="1"/>	<input type="text" value="n"/>
Stav 1 :	<input type="text" value="0"/>	<input type="text" value="3"/>	<input type="text"/>	<input type="text"/>
Stav 2 :	<input type="text" value="1"/>	<input type="text" value="3"/>	<input type="text" value="0"/>	<input type="text" value="0"/>
Stav 3 :	<input type="text" value="2"/>	<input type="text" value="1"/>	<input type="text" value="1"/>	<input type="text" value="1"/>

V riadku 1, stĺpci 4 je nesprávna hodnota

V riadku 1, stĺpci 4 nie je binárny výstup

V riadku 2, stĺpci 3 nie je zadaná hodnota

V riadku 2, stĺpci 4 nie je zadaná hodnota

Opravte všetky problémy a skúste znova

Obr. C.5: Chybové oznámenia vo formulári

Príloha D Obsah disku CD

Na disku CD sa nachádzajú tieto súbory:

- ./Document.pdf – dokument diplomovej práce v slovenskom jazyku vrátane všetkých príloh
- ./Program/ – priečinok obsahujúci zdrojové kódy, ktoré sa spúšťajú bez nutnosti kompilácie
 - ./Program/index.html – stránka štartujúca aplikáciu
 - ./Program/info.html
 - ./Program/neuron.html
 - ./Program/neuronhelp.html
 - ./Program/stav.html
 - ./Program/stavhelp.html
 - ./Program/stavn.html
 - ./Program/stavnhelp.html
 - ./Program/raphael.js – JavaScript knižnica Raphaël
 - ./Program/style.css
 - ./Program/image/ – priečinok obsahujúci 3 obrázky pre úvodnú stránku