



БЪЛГАРСКА АКАДЕМИЯ НА НАУКИТЕ
ИНСТИТУТ ПО ИНФОРМАЦИОННИ
И КОМУНИКАЦИОННИ ТЕХНОЛОГИИ

Кристина Георгиева Капанова

**НОВИ ОПТИМИЗАЦИОННИ СТРАТЕГИИ И
ЕВОЛЮЦИОННИ АРХИТЕКТУРИ ЗА ОБУЧЕНИЕ
НА НЕВРОННИ МРЕЖИ**

А В Т О Р Е Ф Е Р А Т

на дисертация за присъждане на
образователна и научна степен "Доктор"

Научна специалност: Информатика
Професионално направление: 4.6 Информатика и
компютърни науки

Научни консултанти: проф. д.т.н. Иван Димов
и д-р Жан Мишел Селие

София, 2016 г.

Дисертационният труд е обсъден и допуснат до защита на разширено заседание на секция "Паралелни алгоритми" на ИИКТ-БАН, състояло се на 15 ноември 2016 г.

Настоящата дисертация се състои от 6 глави. Дисертационният труд е оформен в 120 страници, включващи 29 фигури, 5 таблици и 180 цитирани източника.

Защитата на дисертацията ще се състои на ... 2017г. от 14:00 часа в зала 218 на Института по Информационни и комуникационни технологии - БАН на открито заседание на научно жури в състав:

1. Акад. Иван Попчев
2. Чл. кор. Красимир Атанасов
3. Проф. д-тн. Иван Димов
4. Проф. Стефка Фиданова
5. Доц. д-р Олимпия Роева

Материалите за защитата са на разположение на интересувалите се в стая 111А на ИИКТ-БАН, ул. „Акад. Г. Бончев“, блок 25А град София.

Автор: Кристина Георгиева Капанова

Заглавие: Нови оптимизационни стратегии и еволюционни архитектури за обучение на невронни мрежи

Обща характеристика на дисертационния труд

Актуалност и съвременно състояние на темата

Обект на изследванията в дисертационния труд са изкуствените невронни мрежи, които могат да бъдат описани като разпределен успореден процесор, който съдържа елементи за обработка на информация. Първоначалните изследвания за изкуствените невронни мрежи са стимулирани от човешкия мозък, който може да бъде разгледан като комплексна и паралелна, информационно-обработваща нелинейна система. Невронната мрежа има две основни сходства с човешкия мозък: знанието, необходимо за функционирането на мрежата е придобито от околната среда в процеса ѝ на обучение, а за съхраняване на тези знания са пригодени синапсови тегла. Методът, използван за процеса на обучение е наричан обучаващ алгоритъм.

Невронните мрежи могат да бъдат както линейни, така и нелинейни. Възможността за нелинейност се отличава с това, че е разпределена по мрежата, и отговаря за начина на формиране на входният сигнал, който също може да е нелинеен. Основно преимущество в работата на невронните мрежи е обучението с учител (supervised learning), при който се описва входната информация в изходна (input-output mapping). За постигането се променят синаптичните тегла на основата на определен набор от учебни примери. В това отношение, всеки учебен пример се състои от входен сигнал и съответстващия му желан отговор. По време на обучителния процес мрежата изменя синаптичните тегла с цел намаляване на отклонението на желания изходен сигнал съгласно предварително избрани статистически критерии. Обучението с учител може да бъде разгледано и като оптимизационна задача.

Способността за адаптиране на теглата на връзките между невроните според околната среда посредством обучителен алгоритъм допринасят за функционирането на мрежите в различни среди благодарение на адаптивността на основните параметри. Тази способност за адаптация е особено важна и при изпълнението на невронните мрежи под формата на електроника. По този начин, дори и при увреждане на определен брой неврони или синаптични връзки, производителността пада незначително. Тяхната успоредна структура също така потенциално може да ускори решението на някои видове задачи.

Качество на невронните мрежи е техният универсален механизъм при обработката на информацията, което позволява една архитектура на невронната мрежа да се използва в множество различни области.

Днес изкуствени невронни мрежи биват прилагани в различни изследователски и приложни сфери. Един от тях е в областта на роботиката, където мрежите се използват като начин да се ориентира манипулациата на

обекти, въз основа на някои сетивни данни. Невронни мрежи се прилагат и за планиране на движения и управление на работи. В областта на машинното зрение невронните мрежи се прилагат за извличане информация за околната среда чрез обработката на данни от колекция на изображения. Невронни мрежи са прилагани със значителен успех във финансовия и банков сектор. Те се използват за измерване на кредитния риск на кандидатите, за процедурата за оценка на кредита и прогнозиране на кредитните ставки, оценка на недвижими имоти, прогнозиране на финансовия пазар и валутния курс, както и за корпоративен финансов анализ.

В сферата на медицината, невронни мрежи са прилагани за анализ на раковите клетки, ЕЕГ и ЕКГ анализ, както и за контрол на качеството в болници.

Цели и задачи на дисертационния труд

Целта на настоящата дисертация е създаване на нов алгоритъм за обучение на изкуствени невронни мрежи, който веднъж изпълнен, позволява на мрежата да излезе от локален минимум или седловинни точки, в които е попаднал алгоритъма. Дисертацията представя и приложение на методи за анализ на чувствителността за изучаване на смущаването на теглата в мрежата, засегнати от шум в системата. Изследването на анализ на чувствителността може да е изключително полезно при хардуерно изпълнение на невронни мрежи, при които шум от различни източници влияе на производителността на невронната мрежа. Дисертационният труд цели и представянето на нов еволюционен алгоритъм за автоматичното генериране на архитектури на невронни мрежи чрез представянето на хибриден генетичен алгоритъм. По този начин, подходи на проба и грешка при създаването на топологии на невронни мрежи се заменя с автоматичното откриване на архитектура на невронни мрежи, подходящи за специфични задачи. Целта е подпомагането на процеса при създаването на изкуствени невронни мрежи и намаляването на човешка грешка в този цикъл.

Конкретните задачи за постигането на тези цели са:

1. Създаването на нов оптимизационен алгоритъм, инициализиран след приключване на обучителния процес на невронната мрежа;
2. Селектиране и практическо приложение на инструменти за анализ на чувствителността при наличие на шум в невронната мрежа;
3. Създаване на нов хибриден еволюционен алгоритъм, с цел автоматизирано изграждане на невронен модел, според определена задача;
4. Програмно осъществяване алгоритмите и инструментите;

Методология на изследването

Методологията на настоящите резултати се базират на изследвания в областите на изкуствения интелект, изкуствените невронни мрежи, оптимизационни методи, както и квантовата физика. Комбинирани са подходи, базирани на закони от физиката, както и на статистически анализ и машинно самообучение.

Използвани са математически индикатори, свързани с анализ на чувствителността при определянето нивото на шум в система, базирана на софтуерно или хладуерно осъществяване на невронна мрежа.

Създаване на хибриден метаевристичен алгоритъм за автоматична конструкция на невронни мрежи, позволяваща в някои случаи избягването на проблема за обобщението на информацията.

Проведени са числени експерименти, чиито резултати са представени и анализирани.

Осъществена е реализация на предложените алгоритми в този дисертационен труд на езика C, част от която е описана в Приложението на разработката.

Апробация на резултатите

Резултати, включени в дисертацията, са докладвани на:

- Семинар на секцията "Паралелни алгоритми 2016;
- Докторантски форум, център за обучение-БАН, 2016; (презентацията за хибридният еволюционен алгоритъм е отличена с награда на форума от организаторите)

Част от резултатите са представени на следните конференции:

- 10th International Conference on „Large-Scale Scientific Computations“ (LSSC'15), Созопол, България, 2015;
- 10th IMACS Seminar on Monte Carlo Methods (MCM 2015), Линц, Австрия, 2015;
- LinuxCon 2015, Дъблин, Ирландия, 2015;

Резултатите от дисертацията са публикувани в 3 статии в международни списания с импакт фактор.

Списък на публикациите по дисертацията

1. K.G. Kapanova, I.T. Dimov, J.M. Sellier, On randomization of neural networks as a form of post-learning strategy, *Soft Computing* (2015). *doi* : 10.1007/s00500 – 015 – 1949 – 1, (IF.1.63).

2. K.G. Kapanova, I.T. Dimov, J.M. Sellier, A Neural Network Sensitivity Analysis in the Presence of Random Fluctuations, *Neurocomputing* (2016), *doi* : 10.1016/j.neucom.2016.10.060, (IF.2.392)
3. K.G. Kapanova, I.T. Dimov, J.M. Sellier, A genetic approach to automatic neural network architecture optimization, *Neural Computing and Applications* (2016), *doi* : 10.1007/s00521 – 016 – 2510 – 6, (IF.1.492)

Съдържание на дисертацията

Настоящата дисертация се състои от 6 глави. Дисертационният труд е оформен в 120 страници, включващи 29 фигури, 5 таблици и 180 цитирани източника.

Глава 1. Увод

В уводната глава на дисертацията е направен кратък исторически анализ на областта на изкуствения интелект, и специфично на невронните мрежи. Описани са основни разлики между невронните мрежи и изчислителните архитектури.

Глава 2. Основи на изкуствените невронни мрежи

Втора глава от настоящата дисертация цели въвеждането в основните понятия и принципи на работа на изкуствените невронни мрежи. Зараждането на идеята и описанието на изкуствена невронна мрежа за пръв път е публикувано през четиридесетте години на 20 век, представяйки опростен модел за работата на човешкия неврон. МакКълук и Питс [22], описват в статията математическата структура на опростен невронен модел, разгледан като прагов логически елемент (ПЛЕ).

В настоящата разработка, под невронни мрежи (НМ) се разбира тематически модел, който се състои от отделни елементи, симулиращи в ограничен модел работата на естествения неврон в мозъка на човека. Невронните мрежи са изградени от отделни елементи, които обработват входящите сигнали, формирайки съответна изходна стойност (изходен сигнал), която се разпространява към елементите, с които е свързан. Можем да определим три основни елемента:

- множество от връзки (синапси), характеризирани посредством теглови коефициент w_{ij} , които могат да приемат положителни или отрицателни стойности.

- суматор, който сумира входните сигнали, умножени по съответния теглови коефициент. Най-широко разпространеният суматор е линейният.
- активационна функция (функция на изхода), чрез която се преобразува получения сумарен вход в изхода (изходния сигнал) на съответния неврон.

В математически вид, функционирането на неврон се описва по следния начин. Определяме $(x_1, x_2, x_3, \dots, x_i)$ като входни сигнали, w_i синаптически тегла на неврона, а w_i като връзка между невроните. Изходните данни (z) на неврона се определят като

$$z = \sum_i w_i x_i \quad (1)$$

и са определени като линейна сумация на входните сигнали и синаптичните връзки. Като част от алгоритъма може да бъде определен и прагов елемент (bias), който се означава със символа (w_0) , и може да бъде описан като:

$$z = w_0 + \sum_{i=1}^n w_i x_i \quad (2)$$

Използването на прага в този смисъл, предпазва ефекта на афинното преобразуване (affine transformation) на изходния линейен суматор.

В глава 2 са описани основните характеристики на невронните мрежи: параметрите на обработващите елементи, типът на входните и изходните стойности, активационните функции, както и топология на мрежата. Най-често срещаните типове активационни функции, които определят изходният сигнал на неврона са функция на единичен скок или прагова функция (threshold function)

$$f(z) = \begin{cases} 1 & : z \geq 0 \\ 0 & : z < 0 \end{cases} \quad (3)$$

както и Сигмоидална функция (sigmoid function), която може да бъде описана по следния начин:

$$y(z) = \frac{1}{1 + e^{\beta z}} \quad (4)$$

Сигмоидала е най-често използван при мрежи с обратно разпространение на грешката, където функцията, която е бързонарастваща, работи за поддържането на баланса между линейно и нелинейно поведение. За преодоляването на определени недостатъци на сигмоидалната функция, може да бъде използвана хиперболична функция (hyperbolic tangent-tanh), която

функционира в по-голямо пространство $[-1, 1]$:

$$f(z) = \begin{cases} 1 & : z > 0 \\ 0 & : z = 0 \\ -1 & : z < 0 \end{cases} \quad (5)$$

Като част от втора глава са описани различни популярни топологии на невронни мрежи, като са представени и илюстрации за дизайна на невронни мрежи. Важна характеристика на невронните мрежи е тяхната мрежова архитектура (структура). Могат да се разграничат няколко фундаментално различни класа от мрежови архитектури. В най-опростения модел, мрежата има входен слой (input layer) на възлите на източника, който предава информацията на изходният слой (output layer) на невроните (изчислителният възел) но не и обратно. Тази мрежа е от право-разпространяващ тип.

Многослойна мрежа Многослойната невронна мрежа е вид невронна мрежа за пряко разпространение на сигнала и се характеризира с наличието на един или няколко скрити слоя (hidden layers), а невроните, групирани в тези слоеве се наричат скрити неврони (hidden neurons) или скрити елементи (hidden units) [3]. Под термина "скрит" се разбира слой или неврон, който не се вижда пряко от входа или изхода на мрежата, като целта на скритите елементи е да допринасят за способността на мрежата да извлече от изхода статистики от по-висок ред. Връзките между невроните във слоевете могат да бъдат описани като насочен граф и обикновено са напълно свързани [12], [6]. Насоченият граф описва не само протичането на сигнала между невроните, но и как сигнала се предава в съответния неврон. Невронната мрежа за пряко разпространение може да бъде описана по следния математически модел, където l обозначава броя слоеве в мрежата, net_j^l описва j -ия неврон в слоя l . Синаптичните тегла от i -ия неврон в слой $l - 1$ до j -ия неврон на слой l са представени като $w_{ij}^{l-1,l}$, а изходната стойност на j -ия неврон в слой l -ия е представена като $f(net_j^l)$ активационна функция:

$$net_j^l = \sum_{i=1}^{n_{l-1}} w_{ij}^{l-1,l} y_i^{l-1}, y_j^l = f(net_j^l) \quad (6)$$

Определянето на броя скрити неврони е от изключително значение при структурирането на невронната мрежа, тъй като прекалено голямо количество неврони намаляват точността на обобщението на данните. Допълнителен проблем се състои от факта, че голямото количество неврони служи за автоматичното запомняне на обучителните данни, което премахва способността за качествено обобщение на информацията. От друга страна, недостатъчен брой неврони допринася за намаляване на обучаващата способност на мрежата и откриването на комплексни гранични стойности

на проблема. Точният брой неврони във мрежата, както и във всеки скрит слой не е обоснован и е строго зависим от специфичната задача, обучителен алгоритъм и от специфични решения на учения.

Общоприет подход към проблема е прилагането на теоремата на Колмогоров [20], в която се посочва, че за изчислението на произволна непрекъсната функция, трябва да се осигури брой неврони, равен на два пъти броя на входните възли плюс още един. Допълнителен метод е изчислението на броя на скритите неврони, описани чрез HM , при наличие на брой неврони във входен слой k и изходен слой v , където

$$HM = \frac{1}{2}(k + v). \quad (7)$$

Рекурентни мрежи За разлика от невронна мрежа с право разпространение на сигнала, рекурентните мрежи имат поне една обратна затваряща се връзка [11]. Този тип мрежи могат да бъдат съставени и само от един слой неврони, които връщат изходните сигнали обратно на входовете на всички неврони в мрежата. При наличие на самозахранваща се обратна връзка, изходът на неврона е върнат обратно на собствения вход.

Дълбоки мрежи Описаните до тук невронни мрежи, са със сравнително опростена архитектура, към която понякога се отнася термина плитка архитектура (shallow architecture), съставена от множество неврони в няколко невронни слоя. Увеличението на наличните данни и данни от по-висок ред предразполага към създаването на дълбоки невронни мрежи (deep neural networks-DNN), които прилагат повече от два скрити слоя на конвенционална многослойна мрежа [24].

В настоящата работа е приложена многослойна невронна мрежа с пряко разпространение на сигнала. Обобщението на този раздел налага и до разглеждането на един от обектите на този дисертационен труд: обучителния процес при невронните мрежи.

Глава 3. Проблемът на обучение на невронна мрежа и изпълнение на нова стратегия след обучение

В глава 3 обучителният процес на невронните мрежи е разгледан като оптимизационен процес и е предложен нов пост-обучителен алгоритъм. Главата е структурирана в два основни раздела. В първия раздел са описани основни оптимизационни алгоритми за обучението на мрежи. Във втората част е представен нов модел на алгоритъм, както и описание на проведените числени експерименти.

В първата част на трета глава са описани основни обучителни стратегии за обучение на невронни мрежи. Основно свойство на невронните мрежи е способността им да се обучават (learn) на основа на предоставени данни, в

резултат на което да подобряват своята производителност по време на процеса посредством определени правила. Съществуват различни алгоритми за обучение, всеки от който, има предимства и недостатъци. Цялостна класификация на методите и приложенията използвани за обучение би била непълна. Различните видове алгоритми могат да бъдат разпределени в две основни групи: обучение с учител и обучение без учител [13]. В първата група попадат алгоритми, които следят за разликата между получения и очаквания изход на мрежата (в този случай очаквания изход на мрежата е предварително зададен от учителя), като итеративно системата извършва корекции на синаптичните тегла съответстващо на предварително избран оптимизационен процес. За разлика от алгоритмите с учител, тези без учител не разчитат на предварителни данни за точния изход на мрежата. Промяната на синаптичните тегла се осъществява по начин, който да позволи на данните да бъдат представени така, че да отговарят на предварително зададен критерий за качеството на представянето и способ на моделиране на параметрите. Могат да бъдат определени няколко последователни стъпки по време на обучителния процес: в невронната мрежа постъпват данни от околната среда в резултат на което се променят свободните параметри на невронната мрежа. Това допринася за изменение на възбуждането на невронната мрежа в следващи стъпки.

В тази глава обучителните стратегии са разгледани като оптимизационен проблем, при който се избира ефективна стратегия за намиране на минималните или максималните стойности на функция, състояща се от много независими променливи. При невронните мрежи тази функция е целевата функция (e.g. cost function, energy or objective function) и пряко зависи от предварителната конфигурация на различни части от мрежата. В следствие, оптимизационния алгоритъм се стреми да достигне най-доброто състояние на системата, в съответствие с обективната функция. Целият процес се повтаря, докато системата е неспособна да намери следващо подобрене на резултата или е достигнала максималния размер на позволените итерации. При обучението с учител, проблемът се определя като минимизиране на функцията, при което се цели грешката да е близост до или равна на нула.

При стартиране на процеса на оптимизация, синаптичните тегла са произволно определени според предварително зададена стратегия. Следващата стъпка въвежда обучаващото множество (x_i, y_i) , където x_i представлява входящ сигнал, а y_i очакван изходен сигнал. Мрежата изчислява решението z , което първоначално значително се различава от желанния резултат, дефиниран чрез y_i . По време на обучителния период, теглото на синаптичните връзки се променя, за да се сведе грешката до минимум.

Описани са няколко често използвани целеви функции, които са описани в таблица) .

Name	Cost Function
L_2	e^2
L_1	$ e $
L_p	$\frac{1}{p} \cdot e ^{\frac{1}{p}}$
<i>Logistic</i>	$\frac{1}{\alpha} \cdot \log(\cosh(\alpha \cdot e))$

Най-използваните функции на грешката. α контролира достоверността на големите различия в стойностите, желания сигнал е обозначен с d , действителното резултат от z и сигнал за грешка като e , при $e = d - z$

Алгоритмите за обучение на невронни мрежи с учител са съсредоточени върху минимизирането на целевата функция за решаването на определена задача. За успешното прилагане на обучителни алгоритми те трябва да функционират с минимална изчислителна тежест. Необходимо е те да са стабилни в присъствието на шум в системата, да са в състояние да произведат резултати, слабо зависими от началните условия. Желателно е изчислителната сложност на алгоритъма да не бъде силно зависима от размерността и размера на данните приложени по време на обучението. Под внимание трябва да се вземат наличие на подходящи критерии за спиране на процеса, за откриването на подходящ резултат и възможна сходимост.

Настоящата глава описва различни видове обучителни алгоритми. Първият от тях е с обратно разпространение на грешката. При този вид алгоритъм, множество итерации са необходими за достигане на решение, в рамките на всяка от които, се разглеждат един или всички примери от обучаващото множество. За всеки такъв пример (x_i, y_i) , се постига право разпространение на активационните стойности и следва обратен анализ на грешките на изхода и разпространение на корекциите на теглата на връзките между елементите от различните слоеве. Този алгоритъм има ниска степен на сходимост, която съответно е и с ниска скорост на обучение, както и необходимост от голям брой обучаващи примери.

При този метод е налична многомерна повърхност за грешки (error surface) при координатите на свободните параметри. Реалната повърхност на грешките се усреднява (averaged) по всички възможни обучителни примери, представени във вид на двойки „вход–изход“ (x_i, y_i) . Повишаването на производителността на мрежата във времето се осъществява при изместване към страната на максимума на повърхността на грешките. Под максимум се разглежда както локален, така и глобален. За да се постигне е необходимо системата да има информация за градиента на повърхността на грешките, който отговаря на текущото поведение на мрежата. Градиента на повърхността за грешки във всяка точка е вектор определящ

най-бързото спускане на тази повърхност [23].

Генетичен алгоритъм - адаптивен евристичен алгоритъм за търсене, основан на еволюционните идеи за естествен подбор [14]. Алгоритъмът представя възможност за използване на случайно търсене в дефинираното пространство на търсене за решаването на даден проблем. Алгоритъмът се стартира с множество от решения, наречени популация. На всяка стъпка, решенията от една популация се вземат и използват от новата популация. Основните параметри на генетичните алгоритми включват вероятност за кръстосване и вероятност за мутация. Първата е свързана с честотата на извършване на кръстосването. Втората взема под внимание колко често части от хромозома ще мутират. Основната функция на мутацията е да предпази алгоритъма от попадане в локален екстремум, но трябва да се прецени честота на използване, тъй като по този начин се променя произволното търсене. Основен проблем е решението за размера на популацията или колко хромозоми да бъдат включени в едно поколение. При липса на достатъчно хромозоми, алгоритъма е с лимитирани възможности за извършване на кръстосване, което води до намаляване на обсега за изследване на пространството на търсене. От друга страна, при прекалено голямо количество хромозоми, алгоритъма ще забави скоростта на сходимост.

Симулирано закаляване - алгоритъма има своето начало от статистическата механика (алгоритъм Метрополис) и използва явна стратегия за избягване от локалните минимума [18]. Алгоритъмът действа чрез симулация процеса на закаляване на метал или стъкло, при което при подходящ план на охлаждане конфигурацията има ниска енергия. За избягването на локалните минимума алгоритъма позволява ходове към решения с по-високи стойности на обектната функция, като такъв ход се счита за “изкачване на хълма” (uphill move). Вероятността за ход към дадена точка се определя чрез

$$Pr[accept] = e^{\frac{-\Delta E}{T}}, \quad (8)$$

където ΔE е разликата между реалната енергия и тази преди хода, и T е ефективната температура на системата. Следователно ходът бива приет при $[0, 1] \ni R < Pr[accept]$.

Други два евристични алгоритъма, използвани за обучителни стратегии, които са разгледани в дисертационния труд включват алгоритъм на мравката [7] и метод на роящите частици [16], [17]. Пример е даден и с квантово симулирано закаляване [15]. Разглеждането на обучението на невронните мрежи като оптимизационен проблем за минимизиране на функцията на грешката, позволява въвеждането на множество локални и евристични алгоритми в областта. Въпреки това, понякога алгоритмите попадат в локален минимум, или в седлови точки, което възпира точното обучение на мрежата.

Нова обучителна стратегия

Втора част на трета глава е посветена на представянето на новия пост-обучителен процес.

В тази част е описан нов пост-обучителен процес, вдъхновен от работата в [2]. Методът имитира квантовите ефекти и по този начин е в състояние да осигури подобряване на теглото на синаптичните връзки в един пост-обучителен процес. Целта е създаването на метод, който да допринесе за нивото на точност на мрежата дори при евентуален неуспех на обучителния процес - дали поради липсващи данни, недостатъчна архитектура или поради попадане на оптимизационния алгоритъм в локален минимум или седловина. Тази нова стратегия е постигната при относително ниска изчислителна тежест. Предложението за прилагане на квантовите механични закони във всеки неврон кореспондира със състоянието на симулация на уравнението на Шрьодингер в зависим от времето метод (или друг еквивалентен формализъм като на Файнман, Вигнър). Това е необходимо, за да бъде числено установен евентуален тунелен преход. Следователно, мрежата трябва числено да симулира следното нестационарно частно диференциално уравнение за всеки неврон:

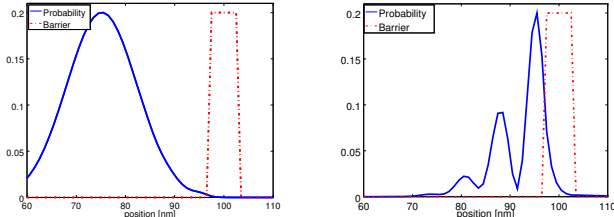
$$i\hbar \frac{\partial \Phi}{\partial t}(\mathbf{r}, t) = \left(-\frac{\hbar^2 \nabla^2}{2m} + V(\mathbf{r}) \right) \Phi(\mathbf{r}, t), \quad (9)$$

където i е имагинерната единица, $\Phi(\mathbf{r}, t)$ е вълновата функция дефинирана чрез пространство и време, \hbar е константа на Планк, r обозначава позицията на частицата, t е времето, m е масата на частицата, ∇^2 е оператор на Лаплас, и $V(\mathbf{r}, t)$ представлява потенциална енергия, която влияе на частицата.

В случай на относително компактни невронни мрежи, състоящи се от един скрит слой и ограничено количество неврони, тази симулация ще бъде изчислително достъпна. В реални приложения на невронни мрежи като тези, използвани за акустичен анализ или разпознаване на реч, в която се използват стотици хиляди неврони, това представлява комплексна и изчислително непосилна задача. Ето защо, представяме изчислително удобна техника, която да имитира наличието на случайността, основна в квантова система, без алгоритъма да страда от високи изчислителни разходи, свързани с наличните и точни квантови симулации.

В този случай, моделираме неврон като хетерогенен полупроводник, който се състои от една енергийна бариера (например AlGaAs), която е разположена между две енергийно по-ниски области (например GaAs). [2] дава възможност за разглеждането на активиращата функция на неврона като една или повече частици, взаимодействащи с бариерата (виж фиг. 1), след навлизането им в хетероструктурата. Вероятността частицата в даден момент t да бъде открита в дадена точка от устройството е описана

от вълновата функция $|\Phi(\mathbf{r}, t)|^2$. По този начин, случайност е въведена в процеса (правило на Борн). Ако вероятността за тунелен преход е по-ниска от вероятността за обратно разсейване, функцията за активиране се счита за инхибиторна на неврона.



Фигура 1 Лява фигура: вълнова частица, описана чрез синя непрекъсната линия, се движи към енергийна бариера, описана в червена прекъсната линия. Дясна графика: след определен времеви период, вълнова частица взаимодейства с бариерата. Част от частицата се разсейва обратно, докато останалата част преминава през бариерата.

В практическо изпълнение, това може да бъде постигнато чрез добавяне към алгоритъма нова функция, описана като **addHiddenNoise**, която да прибави шум към вече изчислените тегла след процеса на обучение. Това се постига само за неврони в скрити слоеве на мрежата.

$$hiddenLayer.neurons[i].weights[j]+ = randomDouble() * weightNoise \quad (10)$$

което отговаря на

$$f(w_1x_1 + w_2x_2 + w_3x_3 + \dots + w_nx_n). \quad (11)$$

Функцията на **RandomDouble** е да ограничи добавянето на шум в определен диапазон, зависещ от спецификата на задачата.

Функцията **backupState** е използвана за копиране на теглата и евентуалното им възстановяване при случаи, в които системата предлага решения, по-лоши от достигнатите.

Числени експерименти

В секция числени експерименти са представени експерименти за валидация на пост-обучителната стратегия. Примери са дадени за две функции: полином от втора степен ($f(x) = x^2$) и корен квадратен на полином ($f(x) = \sqrt{x}$). Системата използва три точки за обучение. Архитектурата

на система е идентична и за двете функции - три слоя - един входен с един неврон, един скрит с 4 неврона, и един изходен с един неврон. Допълнителни органичения са поставени на пространството за търсене на теглото. За първия пример пространството за търсене е ограничено в рамките $-12(min)$ и $+12(max)$, а за втората между $-1(min)$ и $+1(max)$.

Обучителният алгоритъм е симулирано закаляване. В настоящите примери, съзнателно спираме обучението на мрежата в произволен локален минимум посредством скоростта на температурното намаление, като това е достигнато по не-оптимален начин. Целта е да се покаже ясно, че нашата техника може да осигури начин за по-нататъшно подобряване на обучението, дори и след процеса на оптимизация.

За да се разбере степента на влияние на шума върху производството на мрежата и намаляването на обектната функция са изследвани 5 различни сценария (виж фиг. 2). Първо тестваме мрежата без прибавяне на шум, за сравняване на квантовата част от мрежата с класическата част. Останалите експерименти включват работа на мрежата при добавяне съответно на 0.5%, 1%, 2%, и 4% шум.

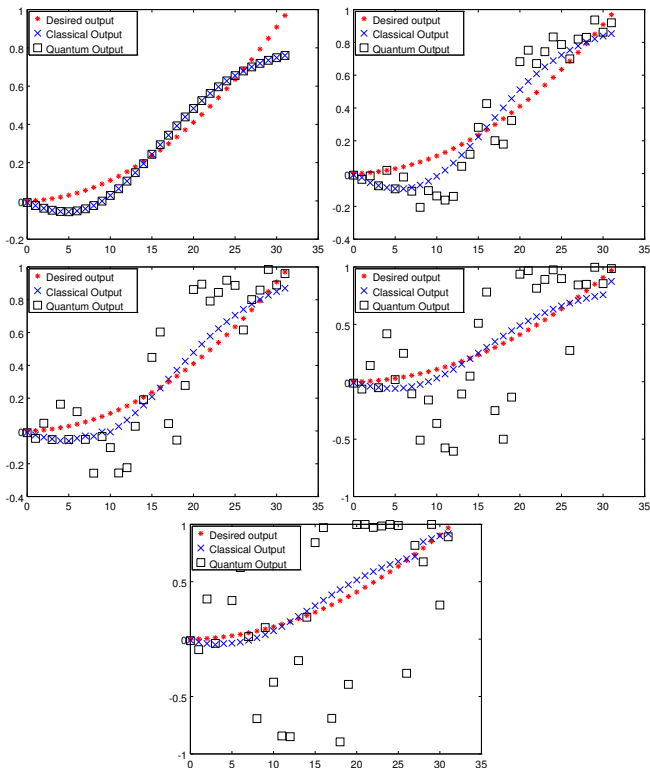
Добавянето на 0.5% шум има минимален ефект върху системата, влияещ на първите няколко резултата. За всяка точка, алгоритъмът сравнява нивото на грешката от класическата и квантовата част, избирайки по-качествения резултат и отхвърляйки другия. При наличие на по-добро квантово решение, мрежата го приема и продължава в следващата итерация. Умереното увеличение на шума до 1% предлага решения от мрежата, които са добри в горната и долната граница на кривата и по-неточни в средните стойности.

Точността на мрежата се влошава при прилагане на 2% шум. Въпреки това, с добавяне на шум, квантовата мрежа превъзхожда класическата в няколко случая (виж фиг. 3 за нивото на грешката). По-високите нива на шум все пак постигат малки подобрения в резултатите на мрежата, но при много по-малка степен и дълбочина.

При втората функция (виж фиг. 4)- за решаване на корен квадратен на полином, се забелязват значителни разлики. Тук, моделът на намаляване на грешката е стъпаловиден, което прилича на този в подобна ситуация за първата функция. При тази настройка, само няколко по-добри резултати са отбелязани, независимо от стръмното спускане на намаляване на грешката от точка до точка.

Симулацията показва, че 2% шум допринася за повишаване на ефикасността на квантовата мрежа. Мрежата осигурява два пъти повече резултати от квантовата мрежа, сравнено с прилагане на 0.5% или 1% шум, като следствие е и много по-ниските нива на грешни решения (виж фиг. 5 за нивото на грешката).

Едно от възможните обяснения за разликата на производителността

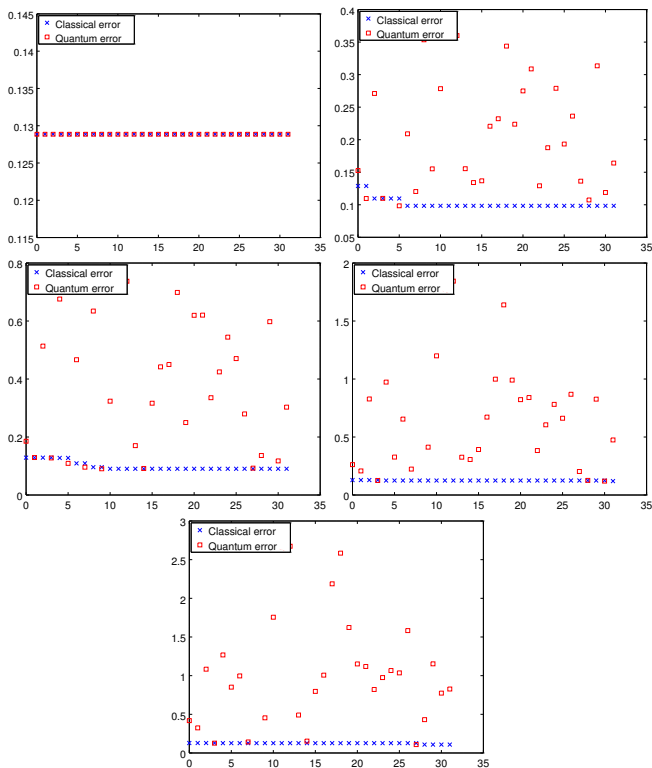


Фигура 2 Графиките се отнасят за решаване на полином от втора степен. Символът червена звезда обозначава желания изходен сигнал на мрежата, синия x показва изходен сигнал от класическата част, а квадрата описва квантовото решение. Първата графика е валидацията тест, при който няма добавяне на шум. Най-горе в дясно са резултатите от мрежата при 0.5% шум. Средните лява и дясна графика описват резултати при прибавяне на 1% и 2% шум. Последната графика е относно резултати при прибавяне на 4% шум.

на мрежата за двата числени експеримента може да бъде предоставена от промяната на границите на пространството за търсене на теглата на връзките при двете функции.

Глава 4. Анализ на чувствителността

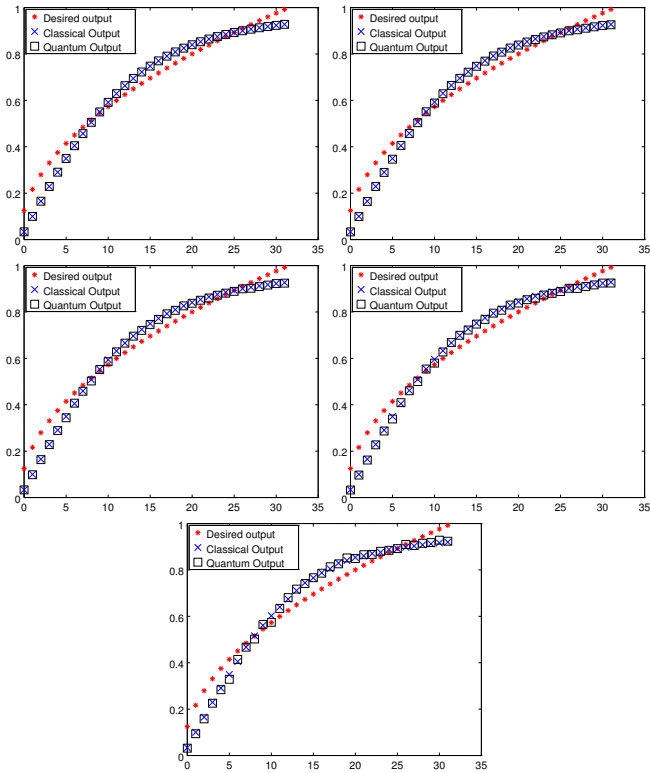
В четвърта глава са представени инструменти за анализ на чувствителността на невронна мрежа, за установяване влиянието на шума в мрежата.



Фигура 3 Намалвяване на грешката в случай на полином от втора степен. Грешката от класическата част от мрежата е изобразен чрез (син) , докато грешката от квантовата страна на мрежата е показан като (червен) квадрат. Най-горе ляво графиката се отнася за грешка при 0% шум с цел валидация. Горната дясна графика илюстрира грешка на мрежата в сценарий с 0.5% шум. Средните графики се отнасят съответно за привабяване на 1% и 2% шум. Последната графика описва грешката на мрежата при 4% шум

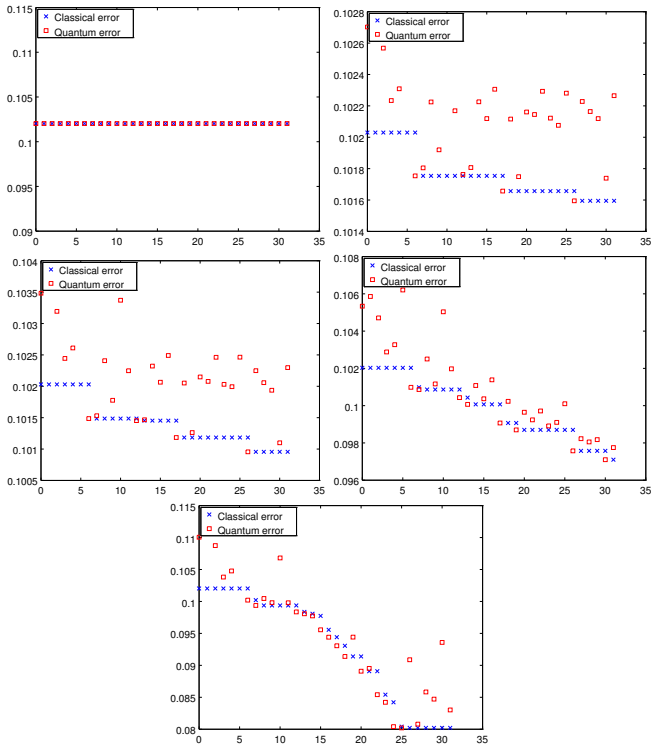
Изследвания за влиянието на случайни или непредсказуеми колебания и смущения и потенциалните им ползи за невронни системи [8, 9] са проведени. От инженерна гледна точка, шумът се идентифицира като вреден за системата и за качеството на резултатите. В биологични нервни системи, от друга страна, шумът е естествено следствие на системата, като се смята, че предоставя определени ползи за анализ на информацията.

В тази глава представяме методология за анализ на чувствителността на невронна мрежа при присъствието на шум, добавен във всяко синапс-



Фигура 4 Графиките се отнасят за решаване на корен квадратен на полином. Символът червена звезда обозначава желания изходен сигнал на мрежата, синия x показва изходен сигнал от класическата част, а квадрата описва квантовото решение. Първата графика е валидация тест, при който няма добавяне на шум. Най-горе в дясно са резултатите от мрежата при 0.5% шум. Средните лява и дясна графика описват резултати при прибавяне на 1% и 2% шум. Последната графика е относно резултати при прибавяне на 4% шум.

тично тегло от мрежата. Целта на тези техники за анализ на чувствителността е да предостави математически подход за определяне на нивата на шум в дадена мрежа. Чрез инструментите установяваме възможно ли е определено количество шум да се използва за подобряване на производителността на мрежата, а когато се влияе негативно - как действа шума като източник на вариации. Чрез инструментите на анализ на чувствителността може да бъде установено влиянието на променливия шум на поведението на мрежата и каква е чувствителността на резултатите от прилагането на



Фигура 5 Намалване на грешката в случай на корен квадратен на полином. Грешката от класическата част от мрежата е изобразен чрез (син) , докато грешката от квантовата страна на мрежата е показан като (червен) квадрат. Най-горе ляво графиката се отнася за грешка при 0% шум с цел валидация. Горната дясна графика илюстрира грешка на мрежата в сценарий с 0.5% шум. Средните графики се отнасят съответно за привабавяне на 1% и 2% шум. Последната графика описва грешката на мрежата при 4% шум

съответното количество шум. Използваните индикатори за анализ на чувствителността показват какво е възможното отклонение на резултатите от системата поради размествания на параметрите на синаптичните тегла. Функционалността на мрежата може да бъде значително променена с изменение на шума в теглата и следователно ще има значителна чувствителност към системата. И обратно, с незначителна чувствителност, производителността на мрежата не трябва да бъдат променени значително.

Използваните индикатори за анализ на чувствителността предоставят

статистически данни, че добавянето на умерено количество шум в действителност съдейства за подобрене на производителността на мрежата. Това се доказва от няколко използвани показатели, които да установят обхвата на възможните решения. Използвани са Евклидовото разстояние (L_2), Чебешева норма (L_∞), и косинусова близост (L_{cos}). При наличие на два вектора на синаптичното тегло, избрани от пространството на възможни решения, косинусова близост предоставя информация за тяхната ориентация, а Евклидово разстояние произвежда възможни посоки и свойства на избрани тегла. L_∞ индикаторът представлява параметър на максимално разминаване. Всеки от изчислителните експерименти е проведен с еднакви начални условия, за да може да се осигури сравнимост на резултатите.

Чрез анализа на чувствителността се цели постигането на по-качествено разбиране относно функционирането на мрежата при въведен шум в пространството на синаптичните тегла. Също така е установена каква е чувствителността на изходните резултати при въвеждането на шум в теглата на връзките. Важно е отбелязването, че шумът е разглеждан като шумова вариация и следователно в някои настройки нивото на шум не води до подобрене на резултатите.

Първият показател, на базата на L_{cos} , се въвежда като

$$l_{cos}(w'') = l_{cos}[u(w''), v] = \frac{u \cdot v}{|u||v|}, \quad (12)$$

където \cdot е продукта и $|u|$ е абсолютната стойност на вектор u . L_{cos} също така може да бъде негативен за вектори, които са в противоположни посоки на N_x пространство на изходните вектори. Това е добър показател за сходство между два вектора. В действителност, когато два вектора сочат в една и съща посока и имат сравними абсолютни стойности, L_{cos} е близо до 1, когато са ортогонални тогава $L_{cos} = 0$, а при противоположност $L_{cos} = -1$. Може да се отбележи, че ако смущаването на тежестите не е твърде чувствителен (даден определен диапазон), тогава L_{cos} трябва да се приближава към 1.

Вторият индикатор, който е използван се основава на евклидовото разстояние и може да бъде описан като

$$I_2(w'') = I_2[u(w''), v] = |u(w''), v|_2 = \sqrt{\sum_{i=1}^{N_x} (u_i - v_i)^2}, \quad (13)$$

където u_i и v_i са i компоненти на вектори u and v . Този показател е метричен и може да се тълкува като разстоянието между началните тегла (u) и изменените след шума (v). Това е ясен показател за качеството на изчисленото решение. По-ниската стойност на L_2 в този случай означава по-добро решение на задачата.

Последният индикатор L_∞ може да бъде приложен за да се определи допълнително метричен показател

$$I_\infty(w'') = I_\infty[u(w''), v] = |u(w''), v|_\infty = \max |u_i - v_i| \quad (14)$$

В този специфичен случай, е използвана стойността L_∞ като индикатор за максималното разстояние, което може да има между изменените тегла след прибавяне на шум в дадена точка от пространственото решение. Като допълнителни единици са представени числовите производни на показателите, въпреки достъпността до тяхното аналитично производно. Целта е постигането на по-добра универсалност.

4.1. Числени експерименти

За анализа на чувствителността, са представени два числени експеримента за изпъкнали и вдлъбнати не-намаляващи функции, с цел избягване изместването на резултатите. Двете функции са полином от втора степен ($f(x) = x^2$) и корен квадратен на полином. И в двата случая архитектурата на невронната мрежа е идентична - с един входящ неврон, 4 скрити неврона и един изходен неврон.

Резултатите от изследването на анализа на чувствителността на невронните мрежи и трите индикатора са изобразени на fig. 6-7. Фигура 6 показва резултатите от тестове за полином от втора степен, където горните графики представляват Евклидовото разстояние, средните графики илюстрират L_∞ и последните графики за показател на косинусовата близост.

Изобразяването на пика на всяка графика представлява оптимално решение на функцията, която отговаря на функция на приложения шум. Важно е да се отбележи, че независимо от вида на индикатора, който е използван, се наблюдава че върховете на възможните решения са представени в една и съща позиция.

В горната лява графика на фиг. 6 евклидовото разстояние показва, че при наличие на 45–47% шум, решението на задачата е значително подобрено. В допълнение, изходите от L_2 и L_∞ са качествено (тяхното поведение) еднакви, въпреки разлика в количествената част. За този конкретен проблем, се очаква тези параметри да предоставят една и съща информация. Може да се отбележи, че малко количество шум представлява достатъчно добро решение на проблема с оптимизацията, въпреки че същото не може да се наблюдава при L_{cos} , където има колебание на решението.

Като се има предвид, че за този конкретен показател, колкото по-близо до 1 е решението, толкова по-добре, графиката на долния ляв участък на фиг. 6 показва влошаване на резултата (движещ се по-далеч от 1). Показателят за косинусовата близост в този вариант е представен с абсолютната стойност $L_{cos} = \left| \frac{u \cdot v}{|u||v|} \right|$. По този начин, независимо от подобни върхове от трите показателя, е възможно да се наблюдава различна информация. Докато първите два показателя демонстрират задоволителни резултати за

нива на шум и поведението на мрежата, третият разкрива обратното заключение.

При евклидовото разстояние (фиг. 6) горна лява графика сочи, че при 40% шум има един връх, който ясно осигурява индикация за подобряване производителността на мрежата. Нивото на шум в този случай драстично подобрява резултатите на невронната мрежа. Третият индикатор осигурява моментна снимка на влошаване на поведението, когато повече шум се прилага. Също така, при намаляващи стойности на L_2 и L_∞ , те показват добро решение на задачата, а при L_{cos} се наблюдават трептения близо до 1, където мрежата произвежда добри резултати, а по-далеч от 1 няма наличие на добри решения.

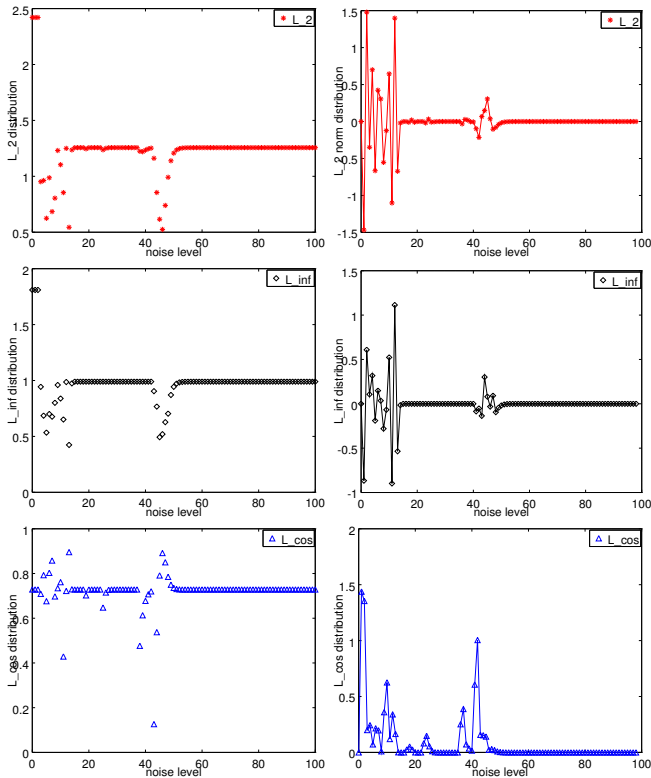
Ситуацията е по-малко ясна в началните нива на шум, когато е налице колебание на оптималните решения. Специално за косинусовата близост, е изчислена втората производна, за да се разбере по-добре свързаните с тях процеси. Долна дясна графика на фиг. 6 на деривата за L_{cos} параметъра, сочи за ситуация, в която по-добри решения могат да бъдат намерени близо до нивото 50% шум, както и в по-малките суми, където шумът е само за 15%.

При втория числен експеримент (виж фиг. 7) за корен квадратен на полином ($f(x) = \sqrt{x}$), трите параметъра и техните деривативни стойности предават подобни наблюдения. Първите два параметъра отново са качествено подобни, въпреки че количествено различни. Наблюдават се пикове за оптималните решения за L_2 и L_∞ , които отново предоставят подобна информация от анализа на чувствителността. Макар че има няколко по-добри решения, получени при по-малко от 10% шум, се наблюдава и появяването на няколко по-добри решения около 70%. Последователно има по-голям шанс за оптимално решение, когато се прибавят малки количества от шум в системата, който подобрява производителността на мрежата (виж фиг. 7, горни и средни фигури). Разликата от предишния случай е, че тук може да се илюстрира, че има няколко малки хълмове сочещи към решение с по-малко колебание от точка до точка. Отново индикатора за косинусовата близост осигурява разнопосочни резултати в сравнение с информацията от предишните показатели. Фигура 7 долу ляво илюстрира този L_{cos} параметър в абсолютната му стойност.

За по-добро разбиране на показателите, отново са използвани производните на показателите, както е обяснено по-горе. Горен лява и средна фигура от фиг. 7 отбелязват че индикаторите гравитират към нулата независимо от количеството шум, въпреки че добри решения са намерени при 5 – 10% шум, както и при 40% и малко над 60% прибавяне на шум.

Всеки един от използваните индикатори допринася важни знания за оптималното решение относно приемливия шум приложен към мрежата. Освен това, с тези инструменти се допринася за установяването на количес-

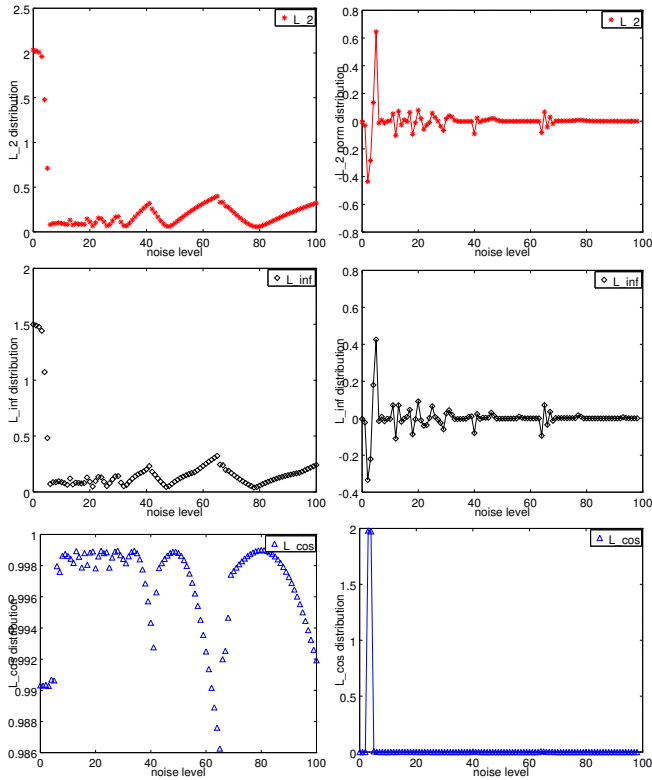
тво шум, което няма да повлияе отрицателно на поведението на мрежата или да има отрицателен ефект върху изходните данни. Нещо повече, тези показатели дават възможност точно да се определи нивото на шума, който предвижда подобрене за процеса на оптимизация, или вместо това действа като източник на колебание.



Фигура 6 Фигурите от лявата страна представляват трите показателя за анализ на чувствителността, приложени за полином от втора степен ($f(x) = x^2$). Графиките от дясната страна описват деривативните стойности на индикаторите

Глава 5. Нов хибриден еволюционен алгоритъм при дизайн на топологии на невронни мрежи

В глава пета са разгледани основните параметри, които влияят на производителността на невронната мрежа[10]- като брой слоеве, брой неврони



Фигура 7 Фигурите от лявата страна представляват трите показателя за анализ на чувствителността, приложени за корен квадрат на полином ($f(x) = \sqrt{x}$). Графиките от дясната страна описват деривативните стойности на индикаторите.

в цялата мрежа, както и във всеки слой, видове синаптични връзки между невроните, видове активационни функции, стратегии за инициализиране на теглана на връзките, обективни функции, обучаващи алгоритми, както и влиянието на предварителната обработка на входни данни към мрежата. Тези компоненти са от съществено значение за изчислителна производителност, ефективност и точност на всяка невронна мрежа[19], [1]. Установено е, че топологията на невронна мрежа влияе върху ефективността на мрежата с примери в областта на компютърното зрение[21]. Броят параметри може да се избира зависимост от проблема и този процес представлява сложен комбинаторен оптимизационен проблем. Различни самоорганизиращи топологии са били предложени и приложени [21], [26], [4], [5].

В настоящата глава е въведен нов подход към развитието на невронна мрежова архитектура, за осигуряването на автоматично и изчислително осъществима самоорганизация на невронни мрежи за решаване на даден проблем. Осъществен е хибриден стохастичен генетичен алгоритъм, който предоставя на мрежата да избере всяка възможна структура от даден интервал от възможни архитектури в зависимост от специфичен проблем. Чрез тази еволюционна стратегия е изграден лесно паралелизиращ се метод в състояние да избяга от проблема с обобщението на изходните данни когато минималното количество на данни за обучение е достъпно.

Описание на алгоритъма Осъщественият алгоритъм предоставя избор на брой неврони, брой скрити слоеве на мрежата, видове синаптична връзка, активационните функции, вид алгоритъм за обучение при избор на топология. Конфигурирането на всички тези компоненти установява многомерно пространство от възможни структури, като същевременно е в състояние да постигне достатъчна производителност с достъпни изчислителни ресурси. По време на числените експерименти, хибридният процес избира между горе описаните елементи.

В настоящата разработка единственото ограничение към броя неврони, слоеве и брой връзки е определен в даден диапазон от минимални и максимални стойности. Няма специфично изпълнение на кода за добавяне или отстраняване на невроните по време на процеса.

Мрежата не се ограничава само до вид връзки между неврони. Алгоритъмът няма ограничение при избор на синаптични връзки между невроните - те могат да бъдат както последователни, така и непоследователни, независимо от позицията на невроните в слоевете. Алгоритъмът предоставя възможност за ограничаване броя на входящите връзки към всеки неврон, зависимост от конкретната задача, която трябва да бъде решена (в зависимост контролирането размера на пространството на решението).

Използваният в тази работа обучителен алгоритъм е симулираното закаливане [18].

Еволюционна стратегия В тази работа, е въведен нов хибрид генетичен подход чрез прилагането на допълнителен стохастичен слой, функциониращ едновременно с генетичния слой. Алгоритъмът включва две синхронни стъпки. Генетичният алгоритъм се инициализира и създава група от популации (50) на отделните архитектури $C_1, C_2, C_3, \dots, C_n$ от пространството на възможните конфигурации. Паралелно с това, стохастичен слой генерира друга група популации (50) на отделните архитектури $S_1, S_2, S_3, \dots, S_n$. Генетичният алгоритъм съчетава вероятност за кръстосване и вероятност за мутация, чрез които се развиват потенциалните топологични комбинации. Следва избор на елементите с най-добри стойности.

Стартирането на алгоритъма определя стойностите на обективната функция за всеки отделен елемент в популацията, което показва, качеството

на специфична мрежова топология. В тази работа, е използвана L_2 – *norm* функция на грешката (въпреки че не е необходимо ограничаването само до тази функция). На всяка времева стъпка, алгоритъмът приема за понататъшна еволюция елемента с най-ниската грешка. Процесът е подобен на Tournament метода [25].

На всяка стъпка, алгоритъмът оценява нивото на грешката на всеки индивид от двете популации. След това избира един индивид от генетичния и стохастичен слой. От там, индивидът с най-ниската грешка е селектиран. Той бива еволюиран стохастично, като се прилагат контролирани настройките описани по-горе.

Допълнителният стохастичен слой води до няколко предимства. Първо осигурява начин алгоритъма да избяга от локален минимум, особено ако цялостното решение има разнообразни популации. Това е осъществено чрез адаптация на минималните стандарти за проектиране в променящата се среда. Стохастичната част позволява на програмата да разработи нова архитектура специфична за даден проблем в достъпно изчислително време дори в многомерно пространство.

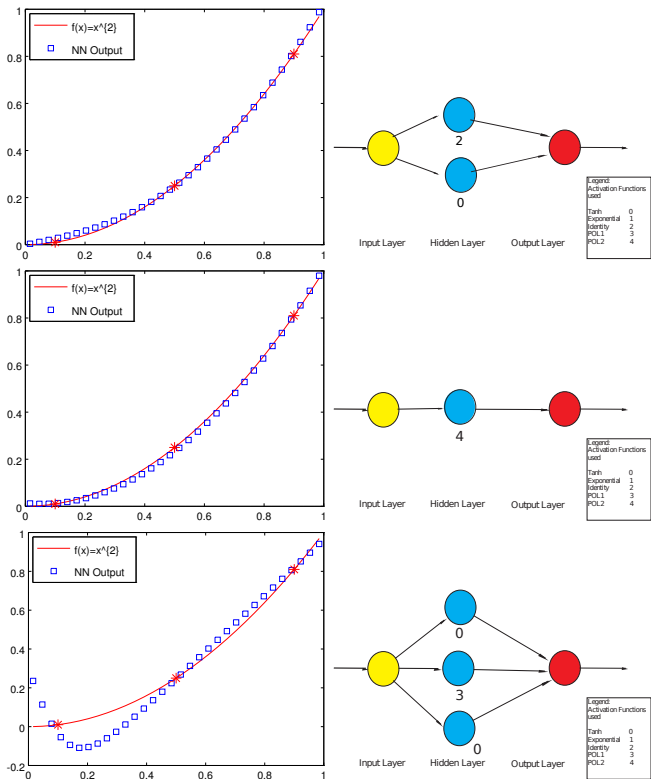
За да бъде прекратен алгоритъма трябва да отговарят на едно от следните две условия. То трябва да достигне максималния позволен брой на поколенията или обектната функция да достигне под определена прагова стойност след r последователни стъпки.

Числени експерименти За да демонстрира възможностите на предложения метод, е извършен числен експеримент за решение на функция $f(x) = x^2$. Предоставени са няколко експеримента в четири сценария с контролиране максималния брой на скрити слоеве и максималния брой неврони. За настоящите експерименти, броя на синаптичните тегла към всеки неврон е ограничен до 2. Тези ограничения се прилагат само за тези конкретни експерименти и не са присъщи за ефективната работа на предложената еволюционен метод.

Според числените експерименти, извършени и дискутирани в този раздел, може да се направи извод, че не всеки път, последователно свързана мрежа осигурява най-доброто решение за определена задача. Когато хибридният алгоритъм използва повече от един скрит слой, често автоматично биват отстранени неврони, което съществено намалява излишни изчислителни ресурси. В тези ситуации, мрежата предоставя достатъчно добри резултати (фигури 8, 9, 10). Интересно наблюдение е, че най-често, невроните в изходния слой използват линейна активираща функция.

В тази глава е въведен нов метод за автоматично търсене на оптимална топология на невронна мрежа за дадена конкретна задача. Чрез реализирания подход са предложени няколко степени на свобода на параметрите, които хибридният алгоритъм може да използва. Следователно алгоритъма произвежда надеждна мрежова топология не въз основа на опита на

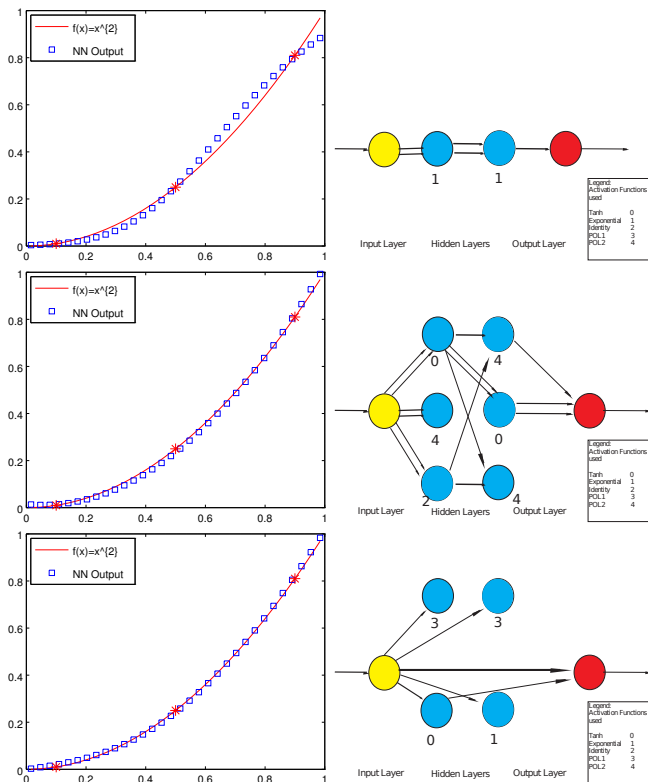
изследователя, но чрез автоматичен метод.



Фигура 8 Графиките в лявата колона представляват изходните резултати на мрежата, докато дясната част описва различните архитектурни решения при ситуация, в която броя на слоевете е 3.

Заклучение

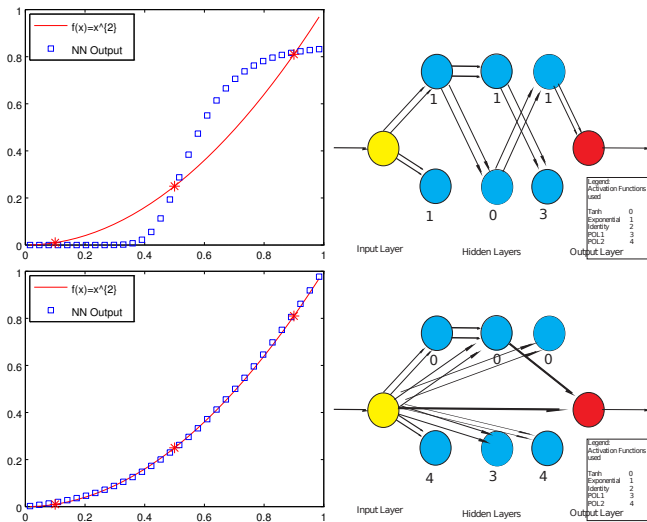
В заключителната част на дисертацията са обобщени методологията и резултатите от изследванията. Новият пост-обучителен модел се въвежда с цел подпомагане на невронната мрежа да излезе от попадане в локален минимум или седловина по време на оптимизационни процес. Въведохме математически инструмент, за да се измери количественото влияние на шума в синаптичните тегла върху работата на мрежата. Въведохме хибриден генетичен алгоритъм за автоматичната еволюцията на невронни мрежови архитектури. Оставени са няколко степени на свобода за проектирането на



Фигура 9 Графиките в лявата колона представляват изходните резултати на мрежата, докато дясната част описва различните архитектурни решения при ситуация, в която броя на слоевете е 4.

мрежата, включително: брой на неврони, брой на скрити слоеве и видове синаптични връзки, използване на функции за трансфер и вид обучаващ алгоритъм.

Авторът се надява, че това изследване ще предостави полезни инструменти за по-нататъшното развитие в областта на невронните мрежи. Описаните резултати са повлияни от квантовата механика. Тези изследвания са само една малка част от бъдещото прилагане на такъв интердисциплинарен подход, както и за практическите приложения на тези методи в практически задачи.



Фигура 10 Графиките в лявата колона представляват изходните резултати на мрежата, докато дясната част описва различните архитектурни решения при ситуация, в която броя на слоевете е 5.

Авторска справка

Изследванията, представени в настоящия дисертационен труд са със следните научни и научно-приложни резултати.

Научно-приложните резултати са формирани предимно от разработването на алгоритми, приложими към обучителния, както и архитектурния процес на изкуствените невронни мрежи:

1. Разработен е и е приложен нов пост-обучителен алгоритъм, осигуряващ системата да не попада в локални минимума или седловини.
2. Въведена е метрика за оценка на влиянието на шума при стойностите на теглата на връзките в невронна мрежа и влиянието на шума за устойчивостта на системата и качеството на резултатите. Направените числени експерименти показват какво е влиянието на шума в мрежата върху ефекта на процеса като цяло.
3. Предложен е нов хибриден генетичен алгоритъм при проблема за автоматичната еволюция на архитектури на невронни мрежи. Методът осигурява търсене в мултимодално пространство за всички възможни решения на проблема даже и при подаване на минимални обучителни данни, намалявайки влиянието на преобучителния процес.

Основните **приложни** приноси на дисертацията са свързани с разработването на софтуер, който прилага новият пост-обучителен алгоритъм, хибридният генетичен алгоритъм и индикаторите за анализ на чувствителността. Софтуерът е разработен на **C** и е пригоден за лесна дефиниция на специфични характеристики при изграждане на топологията на мрежите:

1. минимален и максимален брой на невроните в мрежата,
2. минимален и максимален брой на слоевете в мрежата,
3. вид синаптична свързаност между невроните и слоевете в мрежата,
4. спецификация на различни активационни функции,
5. спецификация на три разновидности обучителни алгоритми - обучение с обратно разпространение на грешката, генетичен алгоритъм, симулирано закаляване.

Част от кода е описан в приложението на дисертацията.

Благодарности

Авторът изказва най-искрени благодарности към своите научни консултанти проф. Иван Димов и доц. Жан Мишел Селие за търпението, разбирането и безкрайната подкрепа, които осигуриха по време на разработването на настоящия дисертационен труд. Авторът изказва и благодарности на екипа от секция "Паралелни алгоритми проф. Стефка Фиданова, доц. Пенчо Маринов, доц. Цветан Остромски, доц. Михаил Недялков и гл.ас. д-р Райна Георгиева за приобщаването на автора към колектива. Благодарности следват и за целия екип на Института по Информационни и Комуникационни Технологии и към неговия Директор за осигурените качествени условия за работа и обучение.

Авторът благодари и на своето семейство за търпението и разбирането по време на изграждането на дисертационния труд, както и на Райна Седова и Николета Георгиева за цялостната подкрепа в БАН.

Литература

- [1] Leandro M Almeida and Teresa B Ludermir. A multi-objective memetic and hybrid methodology for optimizing the parameters and performance of artificial neural networks. *Neurocomputing*, 73(7):1438–1450, 2010.
- [2] Friedrich Beck and John C Eccles. Quantum aspects of brain activity and the role of consciousness. In *How the SELF Controls Its BRAIN*, pages 145–165. Springer, 1994.

- [3] Christopher M Bishop. *Neural networks for pattern recognition*. Oxford university press, 1995.
- [4] Jürgen Branke. Evolutionary algorithms for neural network design and training. In *In Proceedings of the First Nordic Workshop on Genetic Algorithms and its Applications*. Citeseer, 1995.
- [5] Adenilson R Carvalho, Fernando M Ramos, and Antonio A Chaves. Metaheuristics for the feedforward artificial neural network (ann) architecture optimization problem. *Neural Computing and Applications*, 20(8):1273–1284, 2011.
- [6] Gary Chartrand, Linda Lesniak, and Ping Zhang. *Graphs & digraphs*. CRC Press, 2010.
- [7] Marco Dorigo and Mauro Birattari. Ant colony optimization. In *Encyclopedia of machine learning*, pages 36–39. Springer, 2010.
- [8] G Bard Ermentrout, Roberto F Galán, and Nathaniel N Urban. Reliability, synchrony and noise. *Trends in neurosciences*, 31(8):428–434, 2008.
- [9] A Aldo Faisal, Luc PJ Selen, and Daniel M Wolpert. Noise in the nervous system. *Nature Reviews Neuroscience*, 9(4):292–303, 2008.
- [10] Martin T Hagan, Howard B Demuth, Mark H Beale, et al. *Neural network design*. Pws Pub. Boston, 1996.
- [11] Barbara Hammer. *Learning with recurrent neural networks*, volume 254. Springer, 2007.
- [12] Frank Harary. Graph theory. 1969.
- [13] Simon S Haykin, Simon S Haykin, Simon S Haykin, and Simon S Haykin. *Neural networks and learning machines*, volume 3. Pearson Education Upper Saddle River, 2009.
- [14] John H Holland. *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. U Michigan Press, 1975.
- [15] Tadashi Kadowaki and Hidetoshi Nishimori. Quantum annealing in the transverse ising model. *Physical Review E*, 58(5):5355, 1998.
- [16] James Kennedy. Particle swarm optimization. In *Encyclopedia of Machine Learning*, pages 760–766. Springer, 2010.
- [17] James Kennedy, James F Kennedy, Russell C Eberhart, and Yuhui Shi. *Swarm intelligence*. Morgan Kaufmann, 2001.
- [18] Scott Kirkpatrick, C Daniel Gelatt, and Mario P Vecchi. Optimization by simulated annealing. *science*, 220(4598):671–680, 1983.
- [19] Pavel Kordík, Jan Koutník, Jan Drchal, Oleg Kovářík, Miroslav Čepek, and Miroslav Šnorek. Meta-learning approach to neural network optimization. *Neural Networks*, 23(4):568–582, 2010.
- [20] Věra Kurková. Kolmogorov’s theorem and multilayer neural networks. *Neural networks*, 5(3):501–506, 1992.
- [21] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

- [22] Warren S McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133, 1943.
- [23] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning internal representations by error propagation. Technical report, DTIC Document, 1985.
- [24] Jürgen Schmidhuber. Deep learning in neural networks: An overview. *Neural Networks*, 61:85–117, 2015.
- [25] Dirk Thierens and David Goldberg. *Convergence models of genetic algorithm selection schemes*, pages 119–129. Springer Berlin Heidelberg, Berlin, Heidelberg, 1994.
- [26] Xin Yao and Yong Liu. A new evolutionary system for evolving artificial neural networks. *Neural Networks, IEEE Transactions on*, 8(3):694–713, 1997.