Міністерство освіти і науки України
Львівський національний університет імені Івана Франка
Факультет прикладної математики та інформатики

МІЖНАРОДНА СТУДЕНТСЬКА НАУКОВА КОНФЕРЕНЦІЯ З ПРИКЛАДНОЇ МАТЕМАТИКИ ТА КОМП'ЮТЕРНИХ НАУК СНКПМКМ-2019

Львів ЛНУ 2019

Метою конференції є сприяти співпраці між студентами, що вивчають споріднені області прикладної математики та комп'ютерних наук у різних установах. Головні теми: інформаційні технології та системи, моделювання, інженерія та наукові обрахунки.

Організаційний комітет

Дияк Іван - декан факультету прикладної математики та інформатики

Горлач Віталій - заступник декана факультету прикладної математики та інформатики

Онищенко Орест - голова студентської ради прикладної математики та інформатики

Новосад Наталія - студентка факультету прикладної математики та інформатики

Левкович Роман - студент факультету прикладної математики та інформатики

Тарнавський Петро - студент факультету прикладної математики та інформатики

Сенишин Олена - студентка факультету прикладної математики та інформатики

Доскач Денис - студент факультету прикладної математики та інформатики

Та інші члени студенстської ради факультету прикладної математики та інформатики

Спонсори конференції

Зміст

1 ПРОБЛЕМИ ПОБУДОВИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ РЕАЛІЗАЦІЇ ЗАДАЧІ РОСТУ РАКОВОЇ ПУХЛИНИ 7

2 CP-ДЕКОМПОЗИЦІЯ ТЕНЗОРІВ І ПРИКЛАД ЇЇ ЗАСТОСУВАННЯ 10

3 REAL-TIME DECISION MAKING FOR SIMULATED AUTONOMOUS VEHICLES 13

3.1 Introduction 13
3.2 Algorithmic Methods 13
3.3 End-to-end learning based methods 14

4 МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ІМУННОЇ ВІДПОВІДІ НА ВІРУСНУ ІНФЕКЦІЮ ГЕПАТИТУ С 16

5 ВИКОРИСТАННЯ МОНОЛІТНОЇ АРХІТЕКТУРИ ІЗ ПАТТЕРНОМ MVC ПРИ РОЗРОБЦІ ВЕБ-ЗАСТОСУВАНЬ 18

6 ВИКОРИСТАННЯ БАГАТОРІВНЕВОЇ АРХІТЕКТУРИ (N-TIER) ДЛЯ СТВОРЕННЯ ВЕБ-АПЛІКАЦІЇ 20

7 МОДЕЛЮВАННЯ ПОЛІТИЧНОЇ СИТУАЦІЇ В УКРАЇНІ ЗА ДОПОМОГОЮ ТЕОРІЇ ІГОР 23

8 ОБРОБКА ТА АНАЛІЗ ГРАФІЧНО ПРЕДСТАВЛЕНИХ ДАНИХ ТА ЇХ ПРОГНОЗУВАННЯ 27

9 PROPER ORTHOGONAL DECOMPOSITION SURROGATE MODELS FOR SIMULATION AND OPTIMIZATION OF DYNAMICAL SYSTEMS 30

9.1 Introduction and Mathematical Formulation 30
9.2 Application of POD-RBF on Optimal Control Problem of Renewable Resources 31

10 APPLICATIONS OF NEURAL NETWORK MODELS IN KAGGLE COMPETITIONS 33

10.1 Introduction 33
10.2 Results 35

11 CONSERVATION LAWS AND BI-HAMILTONIAN STRUCTURE OF MODIFIED VECTOR DERIVATIVE NONLINEAR SCHRÖDINGER EQUATION 37

12 DEEP NEURAL NETWORK FOR SOUND RECOGNITION 40
12.1 How to do Speech Recognition with Deep Learning 40
12.2 How to build answer with chat bot 40

13 SOLVING PARABOLIC PDES BY MEANS OF STOCHASTIC SIMULATIONS 42
13.1 Introduction 42
13.2 Problem statement 42
13.3 Proposed method 43
13.4 Implementation 43

14 IMAGE COMPRESSION USING MACHINE LEARNING TECHNIQUES 45

15 ЗАСТОСУВАННЯ СПЕКТРАЛЬНОГО МЕТОДУ СКІНЧЕННИХ ЕЛЕМЕНТІВ ДО ОДНОВИМІРНОЇ ЗАДАЧІ АДВЕКЦІЇ-ДИФУЗІЇ. АНАЛІЗ ПОХИБКИ І ПОРЯДКУ ЗБІЖНОСТІ 49
15.1 Постановка задачі 49
15.2 Варіаційне формулювання 49
15.3 Спектральний метод скінченних елементів 50
15.4 Аналіз похибки та порядку збіжності 50

16 ПОБУДОВА ВЕБ-ЗАСТОСУНКУ ДЛЯ РОЗПІЗНАВАННЯ ТА ТРИАНГУЛЯЦІЇ ШКІРНИХ ОБЛАСТЕЙ ЗА ДОПОМОГОЮ ТЕХНОЛОГІЙ NODE.JS ТА VUE.JS 52
16.1 Вступ 52
16.2 Головна ідея 52
16.3 Висновок 54

17 МОДИФІКАЦІЯ ТРИКОВОГО ГРАДІЄНТНОГО МЕТОДУ МІНІМІЗАЦІЇ ФУНКЦІЇ 55

18 КЛАСТЕРИЗАЦІЯ ВИПАДКОВИХ ДАНИХ 57

19 NON LINEAR MIXED EFFECT MODEL BASED APPROACH ON THE PHARMACOKINETICS-PHARMACODYNAMICS OF REMIFENTANIL 58

20 НЕЙРОННІ МЕРЕЖІ ДЛЯ РОЗПІЗНАВАННЯ РУКОПИСНОГО ТЕКСТУ 61

20.1 Загальна постановка задачі роботи із зображенням . . . 61
20.2 Концепція нейронних мереж 61
20.3 Згорткові нейронні мережі (CNN) 62
20.4 Рекурентні шари (RNN) 62
20.5 Згортково-рекурентні мережі для розпізнавання тексту (CRNN) 63

21 ТОЧНІ РОЗВ'ЯЗКИ НЕЛІНІЙНОГО РІВНЯННЯ В ЧАСТКОВИХ ПОХІДНИХ МЕТОДОМ (G'/G)-РОЗШИРЕННЯ

22 ВЕБ-ОРІЄНТОВАНИЙ ДОДАТОК ДЛЯ ВІДСТЕЖЕННЯ ЛАБОРАТОРНИХ РОБІТ TASKER 67

23 ДЕЦЕНТРАЛІЗОВАНІ ДОДАТКИ НА ОСНОВІ БЛОКЧЕЙН 69

24 СТВОРЕННЯ ЧАТБОТІВ, ПОБУДОВА ДИЗАЙНУ ТА ВЗАЄМОДІЇ ЧАТБОТІВ З КОРИСТУВАЧАМИ. ДОСЛІДЖЕННЯ СИСТЕМ РОЗПІЗНАВАННЯ ТА ОБРОБКИ ТЕКСТУ 72

24.1 Вступ 72
24.2 Головна ідея 72
24.3 Головна ідея UI/UX чатботів 73

25 РОЗРОБЛЕННЯ ГРАФІЧНОГО МОДУЛЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ПРОГНОЗУВАННЯ ПОГОДИ 75

26 ЧИСЛОВЕ РОЗВ'ЯЗУВАННЯ ЗАДАЧІ АДВЕКЦІЇ-ДИФУЗІЇ ІЗ ЗАСТОСУВАННЯМ АЛГОРИТМУ ДІРІХЛЕ-НЕЙМАНА 78

26.1 Вступ 78
26.2 Алгоритм Діріхле-Неймана 78
26.3 Вибір пробних функцій 79
26.4 Верифікація результатів 79
26.5 Висновок 80

27 АДАПТИВНА СХЕМА МЕТОДУ ГРАНИЧНИХ ЕЛЕМЕНТІВ ДЛЯ ЗАДАЧА РОПОДІЛУ ЕЛЕКТРОСТАТИЧНОГО ПОТЕНЦІАЛУ 82
27.1 Числові результати 83

28 МІКРОСЕРВІСНА АРХІТЕКТУРА ПРИ РОЗРОБЦІ ТА МОДЕРНІЗАЦІЇ ВЕБ-ЗАСТОСУНКІВ 84

ПРОБЛЕМИ ПОБУДОВИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ РЕАЛІЗАЦІЇ ЗАДАЧІ РОСТУ РАКОВОЇ ПУХЛИНИ

Андрій Кавецький
Україна, Львів
Львівський Національний Університет імені Івана Франка
Факультет Прикладної Математики та Інформатики
andriykavetsky@gmail.com

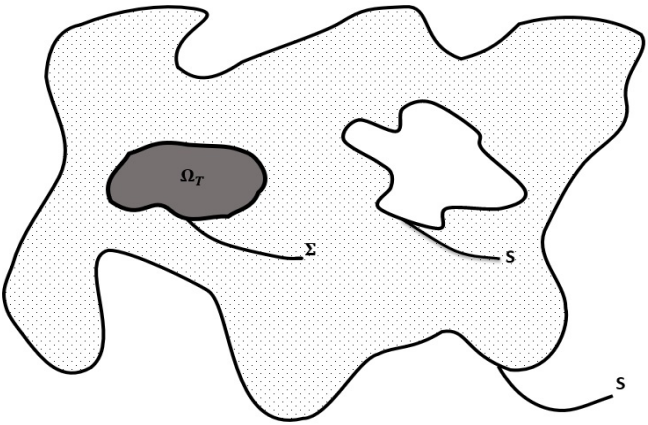


Рис. 1: Досліджувана область з раковою пухлиною.

У роботі розглядається задача чисельного моделювання аваскулярного росту ракових пухлин в живих тканинах (рис. 1.). Математична модель, що використовується для дослідження цього процесу, описана в роботі [1], може бути подана у вигляді початково-крайових задач із змінною границею:

$$0 = \nabla^2 c - c \text{ в } \Omega_T, \tag{1}$$

$$0 = \nabla^2 c \text{ в } \Omega_H, \tag{2}$$

$$c_H|_\Sigma = c_T \text{ на } \Sigma, \tag{3}$$

$$\frac{\partial c_H}{\partial \bar{n}_1} = -\frac{\partial c_T}{\partial \bar{n}_2} \text{ на } \Sigma, \tag{4}$$

$$c = 1 \text{ на } S, \tag{5}$$

де c - шукана функція концентрації, яка залежить від x та t, $D = D_H/D_T$ – відношення коефіцієнта дифузії в здоровій тканині до коефіцієнта дифузії в пухлинній тканині;

$$0 = \nabla^2 p \text{ в } \Omega_T, \tag{6}$$

$$p = k + (G - \chi)(c)_\Sigma - G - AG\frac{x \cdot x}{2d} \text{ на } \Sigma, \tag{7}$$

тут прийняті позначення: p – парціальний тиск, k - локальна повна кривизна, A – коефіцієнт, що описує відносну силу дії апоптозу і мітозу клітин, G – коефіцієнт відносної дії клітинного мітозу до механізмів релаксації(міжклітинної адгезії), d – вимірність простору, x – положення в просторі.
Отримані розв'язки, дають змогу знайти швидкість зміни точок границі пухлини, яка залежить від концентрації, парціального тиску та геометрії області. Ця залежність описується співвідношенням:

$$V = -n \cdot (\nabla p)_\Sigma + Gn \cdot (\nabla c)_\Sigma - AG\frac{n \cdot (x)_\Sigma}{d}, \tag{8}$$

де V - швидкість зміни границі пухлини у напрямку зовнішньої нормалі, n - зовнішня нормаль до границі Σ.

Для чисельного дослідження використовуються схеми методу скінченних елементів, схема Кранка-Ніколсон та схема лінеаризації Ньютона-Рафсона [3]. Побудований програмний комплекс на мові C♯. При створенні програмного забезпечення для подібного класу задач виникає ряд труднощів і аспектів, які потрібно враховувати на етапі програмування, які слід зазначити більш детально:

1. Задання даних для програми. Зрозуміло, що в реаліях ми матимемо справу із тривимірним середовищем і задача розглядатиметься в часі. На етапі наукових досліджень, для верифікації підходів та методів часто розглядають двовимірні задачі. Зокрема зручно мати три форми подачі:

   - Задання границь у вигляді набору точок.
   - У вигляді рівнянь границь поверхонь областей.
   - У вигляді зображень із розпізнаванням областей.

2. Програма мусить передбачати роботу із неоднозв'язними, неканонічними областями, оскільки живі тканини й органи мають неправильну форму. Отже, метод побудови сітки скінченних елементів повинен враховувати ці особливості.

3. Оскільки границі областей дослідження змінюються, потрібно це враховувати при заданні граничних умов. Тому зручно поділити кожну з границь на окремі сегменти, де задаються різного роду граничні умови. Це приведе до того, що для задання границь областей потрібно використати складніші ієрархічні структури для збереження даних. В програмі для збереження даних границь використовується список списків, що дає змогу здійснювати поділ області на підобласті. У подальшому для обчислення граничних інтегралів по області слід враховувати напрям обходу кожної із границь.

4. Для скінченних елементів можуть бути використані різні апроксимації та стратегії. Тому при створенні програмного забезпечення слід використати патерни Абстрактна Фабрика, Будівельник, Фасад, Стратегія, Компонувальник.

5. Задача на тиск та концентрацію розв'язуються окремо, але для знаходження переміщення границі нам потрібно отримати значення в однакових вузлах, тому сітки скінченних елементів для цих задач мають адаптовуватися незалежно від того як ми розв'язуємо ці задачі.

6. Побудова нормалі та відображення сітки. На кожному кроці може відбуватись зміна сітки, але, зрозуміло, що основна частина цих змін буде на примежовому шарі. Тому потрібно реалізувати алгоритм, який дозволить на окремих кроках перебудовувати лише частину сітки.

Мною розроблений програмний комплекс, результати роботи якого будуть продемонстровані під час виступу на конференції.

Література

[1] *V. Cristini* Multiscale Modeling of Cancer. An Integrated Experimental and Mathematical Modeling Approach. / V. Cristini, J. Lowengrub. - Cambridge University Press, New York, 2010. - 278 p.

[2] *V. Cristini* Nonlinear simulation of tumor growth. / V. Cristini, J. Lowengrub, and Q. Nie. - J. Math. Biol, 2003. - 46: - P. 191–224

[3] *Савула Я.Г.* Числовий аналіз задач математичної фізики варіаційними методами. / Савула Я.Г. - Львів, 2004. – 222 с.

[4] *A. Friedman* Analysis of a mathematical model for the growth of tumors / A. Friedman, F. Reitich. – J. Math. Biol, 1999. - 38: -P. 262–284

CP-ДЕКОМПОЗИЦІЯ ТЕНЗОРІВ І ПРИКЛАД ЇЇ ЗАСТОСУВАННЯ

Євстигнеєв Борис Євгенович
Львівський національний університет ім. Івана Франка
Факультет прикладної математики та інформатики
molnerats@gmail.com

У задачах із різних галузей часто виникає потреба працювати з даними, представленими багатовимірними масивами — *тензорами.* Водночас, оперувати ними в «природному» вигляді не завжди зручно та ефективно, особливо у випадку високих *порядків* (кількості вимірів).

Для вирішення подібних проблем застосовують методи *декомпозиції*, тобто представлення складного та громіздкого масиву у вигляді деякого поєднання (суми, добутку) менших. Більшість цих методів є n-вимірними узагальненнями розкладів матриці.

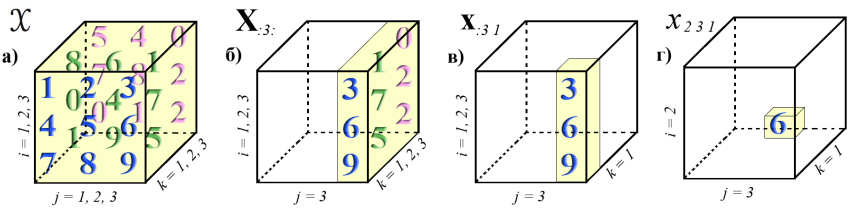


Рис. 1: Тензор 3-го порядку та його індексування

N-вимірний тензор $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$ є **першого рангу**, якщо його можна записати у вигляді тензорного добутку N векторів: $\mathcal{X} = a^{(1)} \circ a^{(2)} \circ \ldots \circ a^{(N)}$. Інакше кажучи, кожен елемент тензора є добутком елементів цих векторів з відповідними індексами:

$$x_{i_1 i_2 \ldots i_N} = a^{(1)}_{i_1} a^{(2)}_{i_2} \ldots a^{(N)}_{i_N}, \quad i_n = 1, 2, \ldots I_n$$

Векторизація та **матрицизація** (або *розгортання за виміром* n, позн. $\mathbf{X}_{(n)}$) представляють тензор у вигляді вектора чи матриці, не змінюючи його елементів і їх порядку. На основі власної програмної реалізації мовою MATLAB досліджено результати різних варіантів оголошення цих операцій та їх сумісність із подальшими обчисленнями.

Ранг тензора $\mathcal{X}$ — це найменша кількість тензорів першого рангу, які в поелементній сумі дають $\mathcal{X}$. Відповідно, **ранговий розклад** —

це представлення тензора у вигляді такої суми. Загалом, визначення рангу є NP-повною задачею.

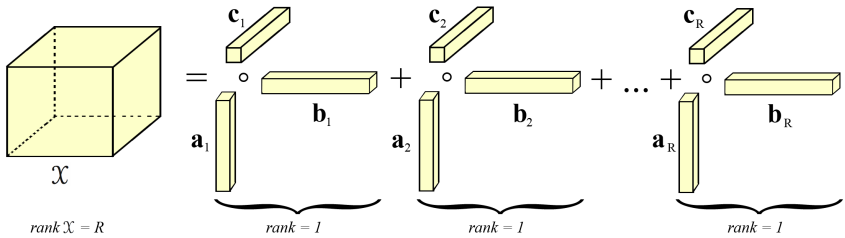


Рис. 2: Ранговий розклад тензора 3-го порядку рангу R

Ідея **CP-декомпозиції** тензорів базується на ранговому розкладі. Для наперед заданого R необхідно мінімізувати норму похибки:

$$\min_{\tilde{\mathcal{X}}} \|\mathcal{X} - \tilde{\mathcal{X}}\|, \quad \tilde{\mathcal{X}} = \sum_{r=1}^{R} \lambda_r \, \mathbf{a}_r^{(1)} \circ \mathbf{a}_r^{(2)} \circ \ldots \circ \mathbf{a}_r^{(N)} \,, \tag{9}$$

де λ_r є необов'язковими коефіцієнтами нормалізації. Якщо вектори $\mathbf{a}_r^{(k)}$ конкатенувати як стовпці $\mathbf{A}^{(k)} = [\mathbf{a}_1^{(k)} \; \mathbf{a}_2^{(k)} \ldots \mathbf{a}_R^{(k)}]$ і скористатись властивостями матричних добутків Кронекера, Хатрі-Рао й Адамара, можна матрично записати алгоритм ALS знаходження розкладу:

function CANDECOMP_PARAFAC($\mathcal{X}, R$)
  for n = 1, ..., N **do** $\mathbf{A}^{(n)} \leftarrow$ матриця($I_n \times R$)
  end for
  repeat
    for n = 1, ..., N **do**
      $\mathbf{V} \leftarrow \mathbf{A}^{(1)\top}\mathbf{A}^{(1)} * \ldots * \mathbf{A}^{(n-1)\top}\mathbf{A}^{(n-1)} * \mathbf{A}^{(n+1)\top}\mathbf{A}^{(n+1)} * \ldots * \mathbf{A}^{(N)\top}\mathbf{A}^{(N)}$
      $\mathbf{A}^{(n)} \leftarrow \mathbf{X}_{(n)}(\mathbf{A}^{(N)} \odot \ldots \odot \mathbf{A}^{(n+1)} \odot \mathbf{A}^{(n-1)} \odot \ldots \odot \mathbf{A}^{(1)})\,\mathbf{V}^{\dagger}$
      нормалізувати стовпці матриць $\mathbf{A}^{(n)}$, $\lambda \leftarrow$ норми
    end for
  until нема покращення результатів **or** максимум ітерацій
  return $\lambda, \mathbf{A}^{(1)}, \mathbf{A}^{(2)}, \ldots, \mathbf{A}^{(N)}$
end function

$\mathbf{V}^{\dagger}$ — псевдообернена матриця $R \times R$, $*$ — поелементний добуток, $\odot$ — добуток Хатрі-Рао.

Вплив нормалізації та різних способів ініціалізації алгоритму на результат вивчено за допомогою як власної реалізації, так і пакету Tensor Toolbox для MATLAB.

Одним із сьогоденних застосувань CP-розкладу є **задача кластеризації документів**. Припускаючи, що: порядок слів у документі не має значення; кожен документ містить R тем t_r, і ймовірність появи слова залежить від теми T; слова в документі незалежні й однаково розподілені — можна записати імовірність появи деякого слова w як

$$P(w) = \sum_{r=1}^{R} P(T = t_r) \cdot P(w \,|\, T = t_r)$$

З іншого боку, $P(w)$ можна обчислити статистично як кількість появ слова, поділену на загальну кількість слів. Проте все одно неможливо однозначно відновити умовні імовірності з цієї формули.

Рішенням проблеми є обчислення імовірності появи трійки слів. Тоді рівняння стане аналогічним до формули CP-декомпозиції, що з урахуванням єдиності цього розкладу за отриманих умов дає можливість обчислити умовні імовірності, і далі за формулою Байєса — $P(T = t_r \,|\, w_1, w_2, w_3)$. Теоретичні засади перевірено на практиці за допомогою Python із бібліотекою TensorLy.

Література

[1] *Tamara G. Kolda, Brett W. Bader.* Tensor Decompositions and Applications. — Society for Industrial and Applied Mathematics, 2009, Vol. 51, No. 3, P. 455-500.

[2] MATLAB Programming. ©1984 - 2004 by The MathWorks, Inc.

[3] *Brett W. Bader, Tamara G. Kolda and others.* MATLAB Tensor Toolbox.

[4] *Nick Vannieuwenhoven.* Tensor decompositions for machine learning applications. — 2017, FWO / KU Leuven.

REAL-TIME DECISION MAKING FOR SIMULATED AUTONOMOUS VEHICLES

Prashant Sinha
Ukraine, Lviv
Ivan Franko National University of Lviv
Faculty of Applied Mathematics and Informatics
prashantsinha94@gmail.com

Introduction

The main research problem this work investigates is the replication of human behaviour in games which involve driving a vehicle, like in Need for Speed or GTA, and designing a system which can mimic the actions of humans in those games and drive the car successfully. This accomplishes a two-fold task. Usually, AI in games is more geared towards games which can be played using a fixed set of rules and logics, like chess or go. However, here we demonstrate the emulation of (nearly) optimal decision making on slightly more complex games, where decisions are not so clear cut and fuzzy, like in NFS. Also, successful implementation will further the cause of autonomous vehicles using only image data, which is the only data available to humans from games. In effect, the goal of this project is to investigate the possibilities of deriving means and methods which can aid decision making for driving games. We shall try and focus on algorithmic methods like lane and road detection, as well as end-to-end learning based methods [1] using Convolutional Neural Networks (CNNs) and present an analysis of which methods provide good results and under what conditions. This kind of investigation has numerous practical applications and benefits. It can lead to systems that can take decisions in complex tasks like games, and demonstrate the effectiveness of artificially intelligent systems that are more general, and not just narrow in their objective. Games are an ideal environment to test complex decision making. Moreover, vehicle automation is an ongoing research problem, which involves both hardware and software aspects. For purposes of safety as well as effective verification of algorithms, it might be better to test on simulations first, and hence this kind of research might give good insights about how to approach such problems.

Algorithmic Methods

A simple but fast algorithmic method that was used was the Hough Line detection algorithm for lane detection. Using the Hough Lines algo-

rithm, we find all the straight lines in the image, and then among these candidate lanes, we choose the longest two lines as the lane lines. This method, despite its simplicity gave surprisingly good results in driving the vehicle autonomously.

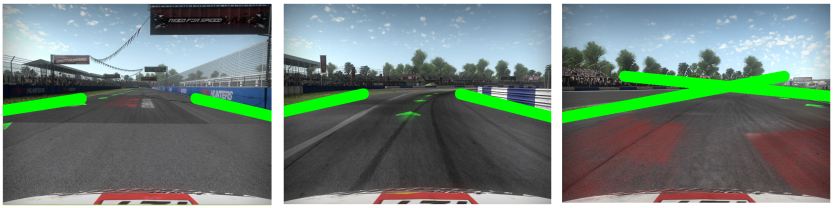


Figure 1: Lane Detection using Hough Lines

However, there are several flaws in this strategy, for example the assumption that the longest two lines will be the lane is not always a perfect assumption. There could be straight lines in the view that are not lanes, such as stands or boundaries or the horizon itself. These problems can lead to wrong lanes be detected.

To address these issues a more advanced lane detection technique was used. This employed techniques such as perspective transform and sliding windows [2]. This method was tried on a previously recorded video to ascertain its effectiveness, and as can be seen from the following figure, the method far outperforms the previous one. But it is significantly slower to use in game at high enough FPS.

Figure 2: Advanced Lane Detection

End-to-end learning based methods

Two different datasets were created by recording a player while playing live, recording both the screen as well as keys pressed. The differences

between the datasets are illustrated in the table below:

	Video Resolution	Dataset Size	Key-Combinations Recorded (Labels)
Dataset 1	80x60	80,000 samples	4
Dataset 2	160x120	400,000 samples	8

Table 1: Dataset Comparison

Both these datasets were balanced. After balancing two different Convolutional Neural Networks were used, namely Alexnet [3] and Inception-v2 [4]. All of these neural networks and dataset combinations gave differing results, which is to be expected. But the combination that performed the best was the one with 160x120 resolution and 4 labels, with a testing set accuracy of 93.28%.

However, it should be noted that testing set accuracy is not the sole criteria to judge the effectiveness of model, as in the real world situation of playing the game, different models presented non-ideal behaviours despite having a good testing set accuracy.

References

[1] *Mariusz Bojarski, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Prasoon Goyal, Lawrence D Jackel, Mathew Monfort, Urs Muller, Jiakai Zhang et. al.* End to End Learning for Self-Driving Cars // arXiv preprint arXiv:1604.07316, 2016

[2] *D.A. Pomerleau* RALPH: Rapidly Adapting Lateral Position Handler // In procs. IEEE Intelligent vehicles Symposium, Detroit ,1995.pages.506-511.

[3] *A. Krizhevsky, I. Sutskever, and G. Hinton* Imagenet classification with deep convolutional neural networks. // In procs. NIPS, Lake Tahoe, 2012

[4] *S. Ioffe , C. Szegedy* Batch normalization: accelerating deep network training by reducing internal covariate shift // Proceedings of the 32nd International Conference on International Conference on Machine Learning, Lille, 2015

МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ІМУННОЇ ВІДПОВІДІ НА ВІРУСНУ ІНФЕКЦІЮ ГЕПАТИТУ С

Олег Українець
Україна,Чернівці,
Чернівецький національний університет імені Юрія Федьковича,
факультет математики та інформатики
ukrainetsoleg@gmail.com

Гепатит С – інфекційне захворювання, спричинене вірусом гепатиту типу С, і передається через прямий контакт із кров'ю інфікованої людини. Вірус гепатиту С заражає клітини печінки - гепатоцити - і розмножується в них. У даний час не існує вакцини проти цього вірусу гепатиту. Тому поряд із клінічними дослідженнями актуальною задачею є математичне моделювання, як процесу поширення захворювання, так і процесу його перебігу.

Відома математична модель імунної відповіді при захворюваності на вірусний гепатит С [1], яка ґрунтується на моделі Г.І. Марчука [2], і задається системою диференціальних рівнянь із запізненням. Також відома модель Водарца, якою описується динаміка конкуренції між вірусом та імунною системою [3,4]. У даній моделі залежно від значення параметрів існує п'ять положень рівноваги. У даній роботі розроблено програму знаходження та аналізу стійкості положень рівноваги, а також динаміки факторів моделі.

Модель Водарца узагальнена шляхом врахування фактору забруднення навколишнього середовища, який може знизити рівень імунної відповіді і ускладнити перебіг хвороби.

Нехай $E(t)$ – усереденений показник забруднення, t – час, де $t \geq t_0$. Якщо фактор E формується декількома чинниками, то

$E(t) = \alpha_1 E_1(t) + \ldots + \alpha_n E_n(t)$, де $\alpha_i \geq 0$ і $\alpha_1 + \ldots + \alpha_n = 1$.

Позначимо через $T(t)$ - кількість неінфікованих гепатоцитів (клітин печінки), а $I(t)$ – інфікованих, $V(t)$ – величину вірусного навантаження, $Z(t)$ – кількість цитотоксичних Т- лімфоцитів (ЦТЛ або CD8) і$W(t)$ – кількість антитіл в організмі. Стан навколишнього середовища не є стабільним, на противагу забрудненню, довкілля має здатність відновлюватись через деякий час $\tau > 0$, і його значення коливається навколо рівноважного стану K>0. Тому вважаємо, що величина E(t) – описується рівнянням Хатчінсона [5]:

$$\dot{E}(t) = r\ (\ 1 - E\ (t - \tau)/\ K\ ) * E(t) \tag{10}$$

де $0 < r$ – коефіцієнт лінійного росту. Із урахуванням впливу фактору

E(t) на підставі моделі Водарца одержимо таку систему рівнянь:

$$\begin{cases} \dot{T} = s - dT - \ \beta VT - \chi E, \\ \dot{I} = \ \beta VT - \ \alpha I - pIZ, \\ \dot{V} = kI - \ \mu V - qVW, \\ \dot{Z} = cIZ - bZ, \\ \dot{W} = gVW - hW \end{cases} \tag{11}$$

Положення рівноваги $E = K$ рівняння (10) асимптотично стійке, якщо $0 < r\tau < \pi/2$.

В узагальненій моделі враховуються, що ріст забрудненості довкілля $E(t)$ веде до росту швидкості інфікування гепатоцитів і відповідно до зменшення швидкості формування неінфікованих осіб. Система рівнянь для дослідження положень рівноваги має такий вигляд:

$$\begin{gathered} s - dT - \ \beta VT - E = 0, \ \beta VT - \ \alpha I - pIZ = 0 \\ kI - \mu V - qVW = 0, cIZ - bZ = 0, gVW - hW = 0 \end{gathered} \tag{12}$$

Досліджено стійкість відсутності захворювання, коли $E = K$, $T = \frac{s-\chi K}{d}, I = V = Z = W = 0$. Таке положення рівноваги досягається і локально асимптотично стійке при виконанні умов:

$$s - K > 0, \ 0 < r\tau < \frac{\pi}{2}, \ \alpha\mu d > K\beta\left(s - \chi K\right).$$

Розроблено програму дослідження динаміки факторів моделі, проведено числове моделювання динаміки систем рівнянь (11), (12) та проілюстровано інші чотири положення рівноваги.

Література

[1] *Романюха,А.А., Руднев С.Г., Зуев С.М.* Анализ данных и моделирование инфекционных заболеваний. Т.2. Математическое моделирование // М.: Наука, 2005. – С. 352-404.

[2] *Марчук Г.И.* Математические модели в иммунологии: вычислительные методы и эксперименты // М.: Наука, 1991. – С. 276

[3] *Jirillo E.* Hepatitis C Virus Disease. Immunobiology and Clinical Applications // Springer, 2008. – С. 123.

[4] *Dahari H., Lo A., Ribeiro R., Perelson A.* Modeling hepatitis C virus dynamics // I. of Theoretical Biology, 2007. – С. 371-381.

[5] *Kuang Y.* Delay Differential Equation with Application in Population Dynamics // Boston: Academic Press. – 1993. – С. 398

ВИКОРИСТАННЯ МОНОЛІТНОЇ АРХІТЕКТУРИ ІЗ ПАТТЕРНОМ MVC ПРИ РОЗРОБЦІ ВЕБ-ЗАСТОСУВАНЬ

Вермій Володимир
Україна, Львів
Львівський національний університет імені Івана Франка
Факультет прикладної математики та інформатики
volodymyr33929@gmail.com

Більшість традиційних застосунків .NET розгортаються у вигляді одного виконуваного файлу, або одного веб-додатку, що виконується в домені додатків служб IIS. Це найпростіша модель розгортання, яка оптимально підходить для безлічі внутрішніх і невеликих загальнодоступних додатків. Проте навіть у такій простій моделі розгортання більшість бізнес-додатків використовує переваги логічного поділу на шари.

Монолітний додаток повністю замкнений в своїй поведінці. Під час роботи він може взаємодіяти разом з іншими службами або базами даних, але в основному його поведінка реалізується в одному процесі, а весь застосунок розгортається як один елемент.

Архітектура застосунку повинна мати як мінімум один проект. В цьому випадку вся логіка знаходиться в одному проекті, також вона компілюється як один елемент.

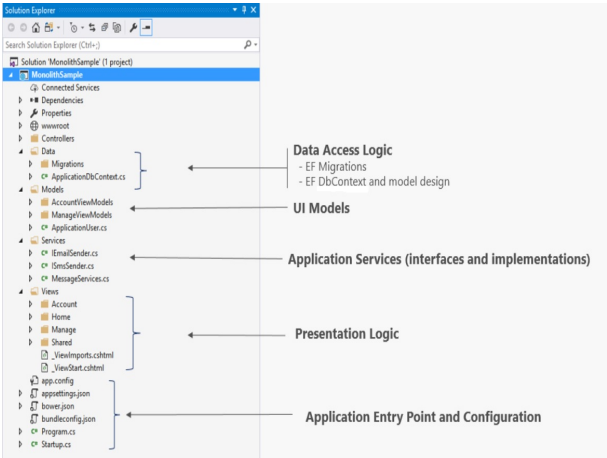


Рис. 1: Монолітна архітектура

Проект ASP.NET Core у Visual Studio на початку представляє собою комплексний монолітний проект. В ньому буде знаходитись уся поведінка додатку, включаючи: бізнес-логіку, логіку-доступа до даних. На рисунку показана файлова структура даного застосунку, який складається з одного проекту.

В сценарії з одним проектом розділ задач реалізується за допомогою папок. Шаблон MVC включає окремі папки (models, view, controller), а також додаткові папки для заданих служб. За такої організації деталі які відповідають за представлення даних максимально можливо розміщуються в папці представлень (Views). Деталі реалізації доступу повинні бути обмеженими класами, які зберігаються в папці (Data). Бізнес-логіка розміщується в службах і класах, що знаходяться в папці моделей (Models).

Переваги:

Головна перевага монолітної архітектури це те, що більшість застосунків використовують одне і те ж – типовий функціонал: логінування, права доступу і т.д.

Коли всі дії відбуваються через один додаток, дуже легко приєднати типовий функціонал до компонентів.

Також є перевага у продуктивності, адже загальний доступ до пам'яті швидший ніж комунікація між процесами (IPC).

Недоліки:

Монолітні додатки сильно зв'язані і стають незрозумілими з "розвитком" застосунку. З часом стає складніше ізолювати сервіси один від одного.

Монолітна архітектура також складніша для розуміння, тому що залежності з часом стають складнішими, і не зрозуміло, який елемент за яку задачу відповідає.

Література

[1] *Adam Freeman* Pro ASP.NET Core MVC 5.

ВИКОРИСТАННЯ БАГАТОРІВНЕВОЇ АРХІТЕКТУРИ (N-TIER) ДЛЯ СТВОРЕННЯ ВЕБ-АПЛІКАЦІЇ

Гутій Тетяна
Україна, Львів
Львівський національний університет імені Івана Франка
Факультет прикладної математики та інформатики
tatiana.gutiy@gmail.com

Сучасні технологічні можливості значно полегшують передачу інформації, підвищують ефективність її обробки та сприяють збільшенню продуктивності роботи програми, в залежності від потреб веб-аплікації розробник вибирає таку архітектурну модель, яка найкраще задовольняє потреби застосунку.

Починаючи розробляти свою веб-аплікацію, було розглянуто багато варіантів побудови її архітектури, але вибір пав саме на багаторівневу архітектуру, зокрема на 3-рівневу, адже її концепція моделі є давно відомою та найпопулярнішою на даний час, вона базується на розподілі всієї системи на окремі ключові функціональні частини та включає в себе шар доступу до даних, шар бізнес правил та шар представлення.

На рис. 1 зображено основні компоненти 3-рівневої архітектури..

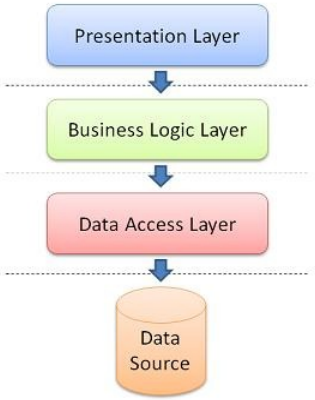


Рис. 1: N-Tier архітектура

Побудувавши аплікацію на MVC ASP.NET, адаптувавши її так,

як показано на рис.2.

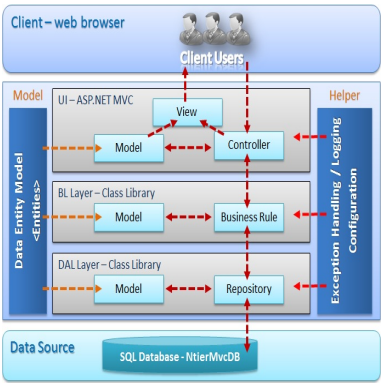


Рис. 2: MVC адаптований до архітектури N-Tier

В процесі розробки було здійснено і аналіз 3-рівневої архітектури та ефективність, в результаті якої були виявленні як переваги так і недоліки, над іншими моделями.

Переваги:

- Високий рівень безпеки
- Масштабованість
- Висока надійність
- Конфігурованість
- Доступність
- Низькі вимоги до продуктивності і технічних характеристик терміналів, як наслідок зниження їхньої вартості

Недоліки:

- Складніша у розгортанні і адмініструванні
- Вища складність створення застосунків
- Складніша у розгортанні і адмініструванні
- Високі вимоги до продуктивності серверів застосунків і сервера бази даних, а, отже, і висока вартість серверного обладнання

Література

[1] *Мартин Фаулер* «Архитектура корпоративных программных приложений» (2002)

[2] *James Charlet* «Enhancing property rates management using N-TIER server architecture»

МОДЕЛЮВАННЯ ПОЛІТИЧНОЇ СИТУАЦІЇ В УКРАЇНІ ЗА ДОПОМОГОЮ ТЕОРІЇ ІГОР

Юлія Пукач, керівник - доц. Гайдучок О. В.
Україна, Львів
Національний університет "Львівська політехніка"
Інститут прикладної математики та фундаментальних наук
ilpach@yahoo.com.ua

Парламентські вибори, вибори президента, місцеві вибори завжди були, є і будуть складним процесом, який потребує значних затрат як зі сторони кандидатів, так і зі сторони виборців.

Оскільки перші мають правильно побудувати та провести передвиборчу кампанію, знайти, розподілити ресурси на підготовку цієї кампанії, а другі, в свою чергу, мають проаналізувати отриману інформацію, оцінити ризики, переваги й недоліки та здійснити важливий вибір,від якого в подальшому залежатиме політична ситуація і загалом життя в країні.

Тому основним завданням даної роботи є здійснити аналіз інтересів громадян та кандидатів, використовуючи моделювання політичної ситуації в країні за допомогою теорії ігор.

Сьогодні дана тема є особливо актуальною, адже це могло б допомогти при побудові передвиборчої кампанії та здійсненні оптимального та раціонального вибору в рамках інтересів сторін, який впливатиме на подальший розвиток дій в країні.

Спираючись на працю [3], в якій розглянуто теоретико-ігрове моделювання суспільно-економічних процесів, а саме теоретико-ігрові моделі для урахування економічних наслідків від прийняття політичних рішень в Росії, побудовано аналогічні моделі на прикладі України.

Основними поняттями даного дослідження є:

Гра – математична модель ситуації, яка характеризується такими ознаками, як 1) наявність декількох (два і більше) учасників; 2) невизначеність поведінки учасників, яка пов'язана з наявністю у кожного з них декількох (два і більше) варіантів поведінки; 3) відмінності інтересів учасників; 4) взаємозалежність поведінки учасників; 5) залежність результату, який отримує кожен учасник, від поведінки інших учасників; 6) наявність правил поведінки, які відомі всім учасникам.

Гравці – сторони, які приймають участь в ситуації і впливають на дії та результати інших учасників.

Рішення, які приймають гравці в рамках правил гри, і відповідно до цих рішень послідовні або одночасні дії називаються стратегіями. Кожному гравцю i відповідає деяка множина стратегій $Y_i, i = \overline{1,n}$.

Набір стратегій, які вибрав кожен гравець, визначає профіль (вектор) стратегій $y = (y_1, y_2, \cdots, y_n)$, який також називають ситуацією гри, де y_i - стратегія і-го гравця. В загальному випадку стратегія гравця - це повний план дій при всеможливих ситуаціях, які можуть виникнути. Від вектора стратегій залежить результат, який отримає гравець.

Гравцю i в кожній ситуації y приписується деяке число, яке виражає ступінь задоволення його інтересів в даній ситуації. Це число називається платежем [1].

Поняття рівноваги характеризує ситуацію в грі, яка задовільняє (принаймні теоретично) усіх гравців. Ситуації рівноваги володіють такою властивістю, що будь-який гравець, який відхиляється від такої ситуації (за умови, що й інші гравці також дотримувались своїх стратегій), не збільшить при цьому свій виграш.

В даному дослідженні використовуються наступні рівноваги:

Рівновага (оптимум) за Парето - ситуація, яка визначається наступним чином. Стратегічна ситуація y^* домінує за Парето ситуацію y, якщо 1)для будь-якого гравця його виграш у ситуації y не більший за виграш у ситуації y^*, і 2)існує хоча б один гравець, для якого виграш у ситуації y^* строго більший, ніж в ситуації y.

Рівновагою за Штакельбергом з лідерством першого гравця називається ситуація, яка визначається наступним чином. Перший гравець вибирає свою чисту стратегію та оголошує про це другому гравцю, а другий гравець вибирає свою чисту стратегію на основі цієї інформації так, щоб мінімізувати свій програш. Таким чином, вибір другого гравця відомий першому гравцю, і перший гравець вибирає свою чисту стратегію так, щоб мінімізувати свій програш. Аналогічно визначається рівновага за Штакельбергом з лідерством другого гравця [2].

Рівновага Неша передбачає таку ситуацію в грі (таку множину стратегій), при якій жодному гравцю не вигідно відхилятися від обраної стратегії, за умови, що інші гравці дотримуються своїх стратегій. Це - ситуація, в якій жоден гравець не може збільшити свій виграш за рахунок власних дій [4].

На основі гри, що розглядається в роботі [3], побудовано теоретико-ігрову модель "нова політична сила (НПС)- виборці".

Поява НПС пов'язана із значним незадоволенням виборців у методах та підходах до правління поточними політичними силами, тобто виборець незадоволений як і владою, так й існуючою опозицією.

Для НПС є дві можливі стратегії: «грати» і «не грати»,а для виборця можливі стратегії: «підтримати» і «не підтримати».

У побудові матричної дискретної гри для такої моделі використовуються дані отримані в результаті соціологічних опитувань, які

проводяться центром «Соціальний моніторинг» спільно з Українським інститутом соціальних досліджень ім. О. Яременко [5],[6], а також у результаті дослідження, проведеного компанією Kantar TNS [7], і ця гра має наступний вигляд:

		Виборець	
		Підтримати	Не підтримати
Нова політична сила (НПС)	Грати	82 22	0 -26
	Не грати	82 -8	0 -7

Виграші НПС розташовані в нижньому правому куті, а виграші виборця - у верхньому лівому. Якщо виборець підтримує НПС, то 1) вибір стратегії «грати» приносить їй позитивний виграш, який описується рівнем підтримки даного кандидата виборцями, а 2) вибір стратегії «не грати» - приводить до програшу НПС, оскільки описує «втрачений» рівень підтримки на момент, коли виборці лише здогадувались про можливість появи нового кандидата. Якщо ж виборець дотримується стратегії - «не підтримати», то 1) вибір стратегії «грати» приводить до програшу, тобто даремно витрачених коштів і ресурсів на проведення виборчої кампанії (і тому він є найбільшим), а 2) вибір стратегії «не грати» теж приводить до програшу, оскільки відчувається «легка досада» щодо нерозсудливості виборців і описується рівнем підтримки поточної влади. Впорядкування виграшів для виборця є наступним : 1) вибір стратегії «підтримати» приводить до виграшу, який описується рівень незадоволення владою, оскільки виборець бажає будь-що змінити політичну ситуацію, а 2) вибравши стратегію «не підтримати», змінити нічого неможливо [3].

Розв'язком даної гри буде набір раціональних стратегій, тобто кожен з гравців прагне шляхом вибору своєї стратегії максимізувати свій виграш. Отже, рівновагою Штакельберга, Неша та рівновагою за Парето для даної гри буде ситуація «грати» - «підтримати» із відповідним виграшем (22,82).

Отже, зважаючи на рівень незадоволення населення поточними політичними силами та виходячи із розв'язків матричної гри, можна зробити висновок, що новій політичній силі вигідно заявити про себе, а виборцям вигідно підтримати її. Таким чином обидві сторони отримують максимально позитивний результат за умови дотримання ними своїх стратегій.

Література

[1] *Дубина І.М.* Основи теорії економічних ігор: навчальний посібник// Барнаул: Вид-тво Алт.ун-ту,2009. - 216с.

[2] *Аркашов М.С., Ковалевський А.П.* Теорія ігор з елементами лінійного програмування: навчальний посібник// Новосибірськ: Вид-тво НДТУ,2016.-98с.

[3] *Шиян А.А.* Теорія ігор: основи та застосування в економіці та менеджменті. Навчальний посібник // Вінниця: ВНТУ, 2009. – 164 с.

[4] *Мулен Е.* Теорія ігор з прикладами із математичної економіки: Переклад з французької// Москва: Мир,1985. 200с.

[5] *Електоральні настрої та проблеми, які найбільш хвилюють населення україни: Осінь 2018* [Електронний ресурс]: [Електронний документ] Режим доступу: https://smc.org.ua/wp-content/uploads/2018/11/PR_2018-11-01_all.pdf.

[6] *Настрої та оцінка загроз українцями* [Електронний ресурс]: [Електронний документ] Режим доступу: https://smc.org.ua/wp-content/uploads/2019/02/RG_3UA_Monitoring_012019_press.pdf

[7] *Індекс задоволеності діяльністю президента, уряду та парламенту* [Електронний ресурс]: [Електронний документ] Режим доступу: https://tns-ua.com/news/indeks-zadovolenosti-diyalnistyu-prezidenta-uryadu-ta-parlamentu

ОБРОБКА ТА АНАЛІЗ ГРАФІЧНО ПРЕДСТАВЛЕНИХ ДАНИХ ТА ЇХ ПРОГНОЗУВАННЯ

Банащук Софія Володимирівна
Україна, Львів
Львівський національний університет імені Івана Франка
факультет прикладної математики та інформатики
sofia.banashchuk@gmail.com

У житті нам часто доводиться працювати з даними, які, зазвичай, подані у графічному вигляді: графіках, таблицях, діаграмах тощо. Така візуалізація, безумовно, є корисною. Проте бувають ситуації, коли дані про якийсь процес подані у вигляді графіка функції, і для того, щоб його проаналізувати та передбачити, як саме він буде розвиватися далі, потрібно спершу отримати числові дані з такої візуалізації за певний проміжок часу. Наприклад, це можуть бути якісь температурні показники погоди чи курси валют. Тому виникає потреба у використанні доступного програмного підходу для обробки такої інформації.

Метою цієї роботи було дослідження та розробка програмного забезпечення, за допомогою якого можна обробляти зображення графіка функції, заданого в декартовій системі координат, та робити певний прогноз щодо процесу, що моделюється, на основі отриманих даних. Поставлену задачу умовно можна поділити на два етапи:

1. обробка зображення (оцифрування графіка функції);
2. побудова прогнозу майбутніх даних.

1. Обробка зображення графіка функції, заданого в декартовій системі координат

Задачі розпізнавання образів, які, в свою чергу, належать до кола задач штучного інтелекту, все частіше виникають при розв'язуванні актуальних проблем у різних сферах життя суспільства. Тому, станом на сьогодні, існує безліч алгоритмів, що широко використовуються при обробці та аналізі зображень.

Для того, щоб виявити власне графік функції, було проведено сегментацію зображення, застосувавши метод Оцу (Otsu's method [1]), основною ідеєю якого є кластеризація за інтенсивністю, після чого було

здійснено пошук найбільшого за площею фрагмента зображення, що і є шуканим об'єктом. Такий підхід був реалізований з використанням досить поширеної бібліотеки комп'ютерного зору - OpenCV [2].

Наступним кроком було розпізнавання чисел (підписів), розташованих на осях Ox та Oy. Перед тим, як застосувати алгоритм оптичного розпізнавання тексту, було визначено два фрагменти зображення, в межах яких має здійснюватися пошук об'єктів. У роботі використано алгоритм з бібліотеки Tesseract, реалізований на основі рекурентних нейронних мереж з архітектурою довгої короткочасної пам'яті (LSTM RNN architecture) [3].

Маючи інформацію про те, що на зображенні є графіком функції $f(x) = y$ та яких значень набувають змінні, необхідно отримати набір конкретних значень функції. Для цього потрібно проектувати точки графіка з певним кроком h, який можна регулювати.

2. Побудова прогнозу майбутніх даних

Числові значення, одержані в результаті попереднього етапу, можна використати для побудови прогнозу деякого процесу, описаного графічно. Для розглянутих прикладів температурних показників та курсу обміну валют за певний період часу було обрано метод простого ковзного (рухомого) середнього (Simple Moving Average [4], [5]):

$$f_k = f(x_k) = \frac{1}{N}\sum_{i=1}^{N} y_{k-i}, \tag{13}$$

де y_{k-i} - реальне значення у момент часу (точці) x_k, N - число попередніх моментів часу, f_k - прогноз на момент часу x_k.

Підсумовуючи вище описане, для створення відповідного програмного забезпечення було використано такі технології:

- мова програмування C++;
- бібліотека OpenCV 2.4.13.6;
- бібліотека Tesseract 4.0.0;
- інструментарій Qt5.

Процес реалізації алгоритмічної частини роботи схематично описаний на рис. 1.

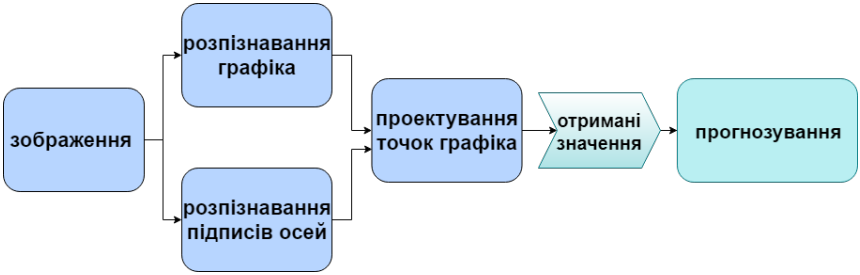


Рис. 1: Схема процесу роботи

Література

[1] *J.R. Parker* Algorithms for Image Processing and Computer Vision // Wiley Publishing, Inc., Indianapolis, Indiana, 2011. - с. 141-142.

[2] *OpenCV (Open Source Computer Vision Library)* https://opencv.org/

[3] *Tesseract Open Source OCR Engine* https://github.com/tesseract-ocr/tesseract

[4] *Сеньо П. С.* Випадкові процеси : підручник // Компакт-ЛВ, Львів, 2006. - с. 36-43.

[5] *В. Ю. Коцюбинський, Л. М. Кислиця* Основи моделювання ринкових ситуацій : навчальний посібник // ВНТУ, Вінниця, 2013. - 99 с.

PROPER ORTHOGONAL DECOMPOSITION SURROGATE MODELS FOR SIMULATION AND OPTIMIZATION OF DYNAMICAL SYSTEMS

Kainat Khowaja
Ukraine, Lviv
Ivan Franko National University of Lviv
Faculty of Applied Mathematics and Informatics
kainat.khowaja@intermaths.eu

Introduction and Mathematical Formulation

This paper employs Proper Orthogonal Decomposition (POD), a surrogate modelling technique integrated in optimization framework for dimension reduction by extracting hidden structures from high dimensional data and projecting them on lower dimensional space. In the first instance, POD is coupled with various Radial Basis Functions (RBF)— a smoothing technique— to provide compact, accurate and computationally efficient representations of ODEs of high-order dynamical systems in environmental economics.

Theoretically, for POD, the function Y(t) is approximated in the domain Ω using a linear combination of some orthonormal basis function $\phi^i(x)$ (constructed with Singular Value Decomposition (SVD) in this paper) such that $Y(t) \approx \sum_{i=1}^{M} a_i.\phi^i(t)$, where the unknown a_i amplitudes are determined by a minimization process of the error of approximation $||Y(t) - \sum_{i=1}^{M} a_i\phi^i(t)||_{L^2}$[2]. Here, the number of basis vectors to be used in approximation is decided by setting a desired error margin ϵ_{POD} and choosing minimum k such that the cumulative energy $E(k) = \sum_{i=1}^{k} \sigma_i^2 / \sum_{i=1}^{d} \sigma_i^2$ captured by first k singular values is less than 1-ϵ_{POD} [1].

Next, RBF is applied to the reduced dimensional space Y_k to continuously interpolate the system responses using an approximation function that is applicable over the entire domain. This results in the approximation $y \approx \phi.B.g(p)$, where ϕ is the basis from SVD, B is the matrix of RBF amplitudes and $g(p)$ is the the function that needs to be calculated for any arbitrary combination of parameters p [2]. This final approximation is called the surrogate model that is constructed using POD-RBF approach. Two types of RBF linear spline:$g_j(p) = ||p - p_j||$ and cubic spline:$g_j(p) = ||p - p_j||^3$ are used for analysis in this paper. The algorithm for application of POD-RBF technique is summarized in figure 1.

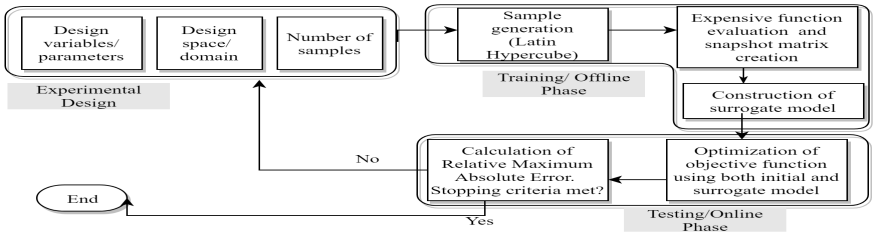


Figure 1: POD-RBF Algorithm Flowchart

Application of POD-RBF on Optimal Control Problem of Renewable Resources

In this section, the performance of POD-RBF method is illustrated on the optimal control problem of optimum exploitation of renewable resources x(t) based on effective utilization, formulated in [3] as following:

$$\begin{aligned} min\ J = -(\int_0^T (p-c)u(t)e^{(-\rho t)}(1-ae^{-bt})dt), \\ x'(t) = rx(t)(1-\tfrac{x(t)}{N}) - u(t), \\ x(0) = x_0,\ x(T) = x_T;\ 0 \leq u(t) \leq \bar{u} \end{aligned} \tag{14}$$

Here, p is revenue per unit in \$ per ton, c is cost per unit in \$ per ton, u/$\bar{u}$ is exploitation amount in tons (control function), ρ is instantaneous social rate of discount, a and b are derived from effective utilization rate, r is intrinsic growth rate, and N is carrying capacity of ecosystem in tons. The initial values of these parameters for numerical experiment are set to be $[p, c, \bar{u}, \rho, a, b, r, N] = [38, 20, 20000, 0.1, 0.4, 0.05, 4.4, 100000]$. The initial condition $x(0)$ is 5000 and the optimization constraint defined at final time $T = 1$ is $x(T) = 15000$.

Latin Hypercube sampling is used to generate various number of sampling points n_s for u(t) (interpolated using $n = 2$ optimization parameters) at $n_t = 100$ time instances. The tolerance is $\epsilon_{POD} = 0.01$ and the number of test points, for which the model was tested, $n_g = 10$. The relative information criteria $E(k) \leq 1 - \epsilon_{POD}$ led to POD basis of dimension $k = 4$ for various n_s and RBF techniques. The system in 14 was then solved for n_g points using both original and surrogate models and the errors are summarized in Table 1. The system responses for one test point are illustrated in Figure 2. Table 1 shows that among all the surrogate models that were trained using different number of sample points, the cubic spline RBF showed the lowest error for the state variable x. Also,

as expected, the error of approximation shows a decreasing pattern as the number of sampling points increase from 60 to 80.

RBF	n_s = 40	n_s=60	n_s= 80
Linear	0.05917	0.05093	0.01346
Cubic	0.02754	0.02336	0.01014

Table 1: Relative Maximum Absolute Errors(RMAE) of POD-RBF

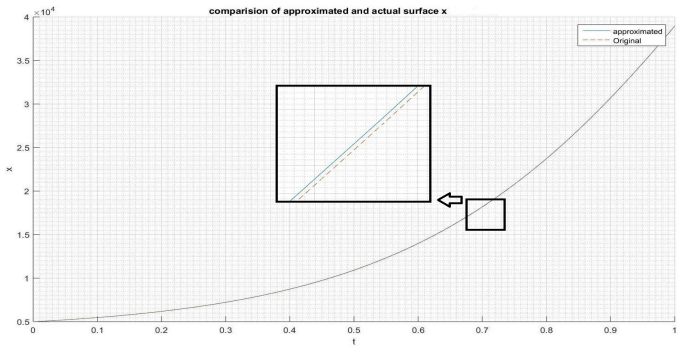


Figure 2: Original vs Approximated System Responses

For the final step, the constrained optimization problem in 14 was solved and timed using initial and surrogate models and RMAE was calculated for the test points. The result showed that the RMAE was only 1.294138e-02 and average computational time was 20sec less for optimization through surrogate models. Hence, a reduced 4-dimensional POD-RBF model provided a highly accurate and computationally efficient parameterization of high dimensional system.

References

[1] *Mykhaylo Shcherbatyy and Iryna Shcherbata* POD for Ordinary and Partial Differential Equations// Prague, International Conference PDMU, 2018.

[2] *Vladmir Bujlak* Inverse Analysis with Model Reduction (Proper Orthogonal Decomposition in Structural Mechanics) // Milan, Springer, 2012.

[3] *Rui Wu, Zhengwei Shen and Fucheng Liao* Optimal Control of Renewable Resources Based onf the Effective Utilization Rate // Hindwai, 2015

APPLICATIONS OF NEURAL NETWORK MODELS IN KAGGLE COMPETITIONS

Grace Anulika Eze, Ukraine/Italy
Ivan Franko National University of Lviv/ University of L'Aquila
Faculty of Applied Mathematics and Informatics/DISIM
Intermaths program, Interdisciplinary mathematics
grace.eze@intermaths.eu
Supervisor: Vasyl Vavrychuk

Introduction

This paper focuses on implementing and analyzing a neural network model on the "Malaria Cell Images Dataset" from Kaggle which contains over 27,588 blood smears image patches from patients which is equally separated into two distinct classes since there is no training label file. The difference between the uninfected and the parasitized red blood cells is that certain regions in varying locations of the cell contains the malaria parasite for the parasitized cells. We implemented a Convolutional Neural Network(CNN) with the Keras framework which we use for the medical image classification. CNN's are biologically inspired deep feedforward , backpropragate neural networks which are sparsely connected to the input layer and are designed to process pixel data of an image in an alternating manner till specific features of the problem image is learned.

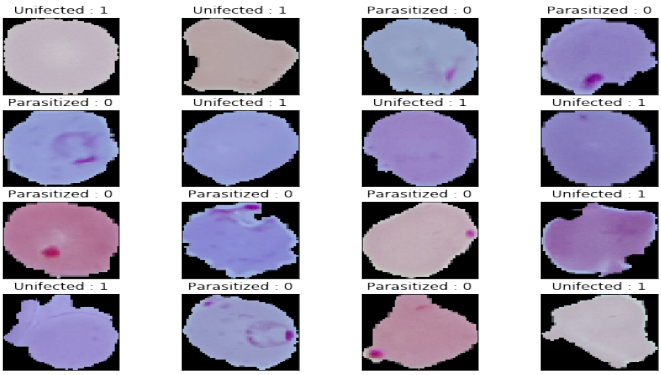


Figure 1: Cells from the Pre-processed Malaria Dataset

Considering a number of m training image samples, a given input image x which consists of an array of pixel intensities has input dimensions $x \in \mathbb{R}^{h \times w \times d}$ where h is the height, w is the width and d is the number of input channels. We are also given p number of filters, k with dimension $k \in \mathbb{R}^{f \times f \times d \times p}$ and for every filter a bias b having dimensions $b \in \mathbb{R}^p$. For the convolution layer, the input image from the sample is convolved with p filters where each filter aims to extract and learn specific features from the image. The filter is passed in equal and finite strides over local regions(receptive fields) in the image and at each location computes the sum of the element-wise dot product between it's element and the elements of the connected receptive field as well as adds the bias per filter. Every computation produces p extracted feature map of the input image as our output for the next layer. Mathematically, the process which generates the output of a convolution layer is given by

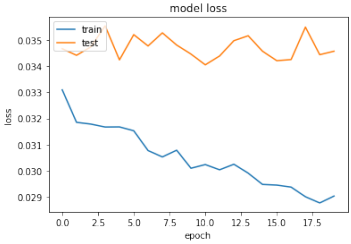
$$(x * k)_{ij} = \sum_{m=0}^{f-1} \sum_{n=0}^{f-1} \sum_{l=1}^{d} k(m,n)x(i-m, j-n) + b \tag{1}$$

$$= \sum_{m=0}^{f-1} \sum_{n=0}^{f-1} \sum_{l=1}^{d} k_{m,n,l} \cdot x_{i+m,j+n,d} + b \tag{2}$$

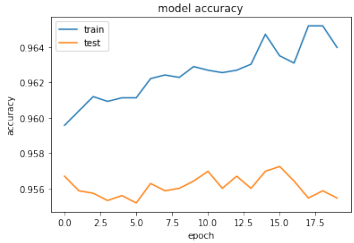
$$= \sum_{m=0}^{f-1} \sum_{n=0}^{f-1} k_{m,n} \cdot x_{i+m,j+n} + b \quad if \quad \#d = 1 \tag{3}$$

Then an activation function is then applied on the output of the convolutions to introduce non-linearity into the network. We use the ReLU activation to remove all non negative entries and replace them with 0.

For the pooling layer, the spatial size of the output gotten from the activations is reduced by reducing the amount of parameters and computation in the network without much loss of significant information. The output matrix is then vectorized and sent to a fully connected layer with a Softmax/Sigmoid classifier which trains and classifies the images using Back-propagation and predicts the label class of the input image.



(a) Loss graph

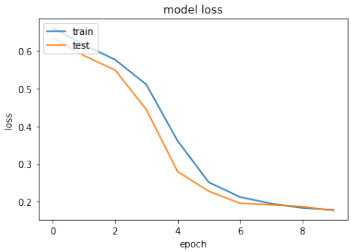


(b) Accuracy graph

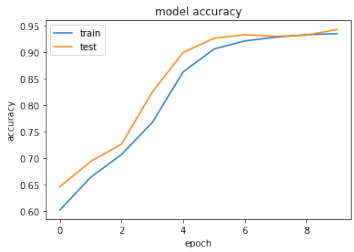
Figure 2: The Loss and Accuracy for the first training of the Malaria Dataset

Results

In Figure 2, the graphs showed some overfitting although the accuracy on the train was about 96% while the accuracy for the validation set was very low, therefore an $L1-$regularization parameter, reg $= 0.001$ wass introduced to correct the overfitting.



(a) Loss graph



(b) Accuracy graph

Figure 3: Visualizing the Loss and Accuracy for the Malaria Dataset

The accuracy after training the model after a number of iterations with a stochastic gradient descent optimizer and a categorical cross entropy loss function was 0.9348 and the loss was 0.2135. The validation accuracy was 0.9365 and validation loss as 0.2149. After computing the confusion matrix on the test sample to determine the performance of our classification model, we have that:

		Predicted		
		Negative	Positive	Total
Actual	Negative	TP: 2512	FP: 248	2760
	Positive	FN: 177	TP: 2575	2752
	Total	2689	2823	5512

Table 1: Confusion Matrix

class	Precision	recall	f1-score	support
0	0.93	0.91	0.92	2760
1	0.91	0.94	0.92	2752
average	0.92	0.92	0.92	5512

Table 2: Classification Report

References

[1] *Fouzia Altaf, Syed M. S. Islam, Naveed Akhtar, Naeem K. Janjua* Going Deep in Medical Image Analysis: Concepts, Methods, Challenges and Future Directions // February 2019.

[2] *Anthony L. Caterini* A Novel Mathematical Framework for the Analysis of Neural Networks. // Ontario, Canada, 2017.

[3] *Andres Calderon, Sergio Roa and Jorge Victorino* Handwritten Digit Recognition using Convolutional Neural Networks and Gabor filters // Proceedings of the International Congress on Computational Intelligence CIIC, 2003.

CONSERVATION LAWS AND BI-HAMILTONIAN STRUCTURE OF MODIFIED VECTOR DERIVATIVE NONLINEAR SCHRÖDINGER EQUATION

Ivan Mykhailiuk
Ukraine, Lviv
Ivan Franko National University of Lviv, Ukraine,
Faculty of Applied Mathematics and Informatics;
University of L'Aquila, Italy,
Dep. of Information Engineering, Computer Science and Mathematics
iv.mykh@gmail.com

Ever since people realized that most of the real-world systems are inherently nonlinear in nature, dealing with this kind of phenomena has been one of the most challenging tasks for science. This is why the theory of *nonlinear dynamical systems* is an important branch of contemporary mathematics, and its problems and results are of interest for both "pure" mathematicians and scientists from other fields.

Some of the key notions of the theory are those of *the complete integrability* and the related class of *completely integrable systems.* Whereas most of the nonlinear systems are not integrable (and usually manifest chaotic behaviour), the study of completely integrable systems gives us an insight into the nature of many phenomena. In particular, integrable systems yield exact solutions for many problems of very advanced modern mathematics and theoretical physics. [4]

In paper [1] Willox et al. established the bi-Hamiltonian structure and constructed the recursive operator for the following *modified vector derivative nonlinear Schrödinger (MVDNLSE)* equation

$$w_t = \begin{pmatrix} u \\ v \end{pmatrix}_t = \begin{pmatrix} -u\left(u^2+v^2\right) - \beta u + v_x \\ -v\left(u^2+v^2\right) - u_x \end{pmatrix}_x = K\left[w\right] = K\left[u,v\right], \qquad (15)$$

where $K : M \to T\left(M\right)$ is a Fréchet smooth polynomial vector field which is defined on infinite dimensional functional manifold $M = C_l^\infty\left(\mathbb{R}^1;\mathbb{R}^2\right)$, $\mathbb{R}_+^1 \ni l < \infty$ is a period, $t \in \mathbb{R}^1$ is the evolution parameter.

In the present work, we investigated Eq.(15) for the complete integrability utilizing a different methodology, namely *the gradient-holonomic* and *the small parameter methods* [2, 3].

Firstly, by means of the gradient-holonomic method we have constructed *the infinite hierarchy of conservation laws* (integrals of motion), which is a necessary condition for the system to be completely integrable (in the Liouville sence). Hence, first three conservation laws for Eq. (15)

have the following form:

$$\gamma_0 = i \int\limits_{x_0}^{x_0+l} u^2 + v^2 + \frac{\beta}{4}\, dx; \tag{16}$$

$$\gamma_1 = \frac{1}{2}\sqrt[4]{-1} \int\limits_{x_0}^{x_0+l} \left(u^2+v^2\right)^2 + \beta\left(u^2-v^2\right) - 4uv_x + \frac{\beta^2}{16} - C_1\, dx; \tag{17}$$

$$\begin{aligned}\gamma_2 = \int\limits_{x_0}^{x_0+l} & \frac{1}{2}\left[(v_x)^2 - (u_x)^2 + \beta u_x v\right] - \frac{1}{3}\left[(1+4i)\, u^3 v_x + (1-4i)\, v^3 u_x\right] + \\ & + \frac{i}{2}\left[\left(u^2+v^2\right)^3 + \frac{1}{4}\beta\left(u^2+v^2\right)^2 + \beta\left(u^4-v^4\right) + (u_x+v_x)^2\right] + \\ & + \frac{i}{4}\left[\beta^2 u^2 - 2\left(u^2\right)_x v^2 - 2C_2\right]\, dx.\end{aligned} \tag{18}$$

Secondly, utilizing the small parameter method we have rediscovered *the implectic pair of Noether operators* of the form

$$\vartheta = \begin{pmatrix} \partial & 0 \\ 0 & \partial \end{pmatrix}, \tag{19}$$

$$\eta = \begin{pmatrix} \beta\partial + 2u_1\partial u_1 - 2u_{1,x}\partial^{-1}u_{1,x} & 2v_1\partial u_1 - 2u_{1,x}\partial^{-1}v_{1,x} \\ \partial^2 + 2u_1\partial v_1 - 2v_{1,x}\partial^{-1}u_{1,x} & 2v_1\partial v_1 - 2v_{1,x}\partial^{-1}v_{1,x} \end{pmatrix}, \tag{20}$$

which exactly coincides with the one given in [1].

Hence, for system (15) the infinite hierarchy of conservation laws and the implectic pair of Noether operators were found. Furthermore, it was shown that the obtained results agree with ones derived by means of a different methodology in [1].

References

[1] *Willox R.* Complete Integrability of a Modified Vector Derivative Nonlinear Schroedinger Equation. / *R. Willox, W. Hereman and F. Verheest* // Physica Scripta, Vol. 52. – 1995. – p. 21–26.

[2] *Hentosh O. Ye.* Differential-Geometric and Lie-Algebraic Foundations of Investigating Nonlinear Dynamical Systems on Functional Manifolds / *O. Ye. Hentosh, M. M. Prytula, A. K. Prykarpatskyi* – Lviv: Publishing center of the Ivan Franko National University of Lviv, 2006. – 408 p.

[3] Integrable dynamical systems: spectral and differential geometric aspects / *Y. A.Mitropolskiy, N. N. Bogoliubov, A. K. Prikarpatskiy, V. G. Samoilenko.* – Kyiv: Naukova dumka, 1987. – 296 p.

[4] *Szablikowski B.* Geometric aspects of integrable systems / *B. Szablikowski* – Oprava: Mathematical Institute of the Silesian University, 2008. – 59 p. – Lecture notes.

DEEP NEURAL NETWORK FOR SOUND RECOGNITION

Ivan Yuzvyshyn
Ukraine, Lviv
Ivan Franko National University of Lviv
Faculty of Applied Mathematics and Informatics
vanuy1305@gmail.com

How to do Speech Recognition with Deep Learning

The first step in speech recognition is simple - we need to convert sound waves into some numeric representation. Sound waves are one-dimensional, it means that at some moment in time, we have a single value based on the height of the wave. To turn sound wave into numbers, we just record the height of the wave at equally-spaced points this method is called *sampling*. To make this data easier for a neural network to process, we are going to break apart this complex sound wave into chunks. Then by adding up how much energy is in each of those frequency bands (from low to high), we create a fingerprint of sorts for this audio snippet.

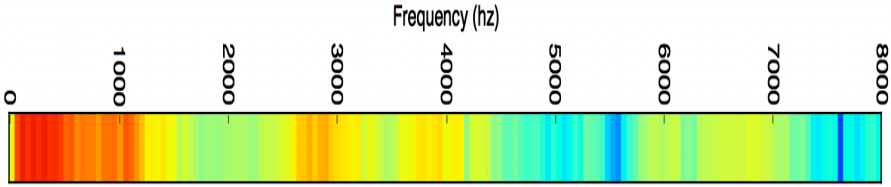


Figure 1: Our 20 millisecond sound snippet

After we run our entire audio clip through the neural network (one chunk at a time), we'll end up with a mapping of each audio chunk to the letters most likely said during that chunk.

How to build answer with chat bot

Our aim is to write bot which will be able to build a proper answer for question ginve in sound wave from first paragraph. But before we need to teach our neural networabout k what to answerIn teaching wasas used simple set of question and answers. The question and answers are joined to extract the total vocabulary, also we need to convert all words/characters

into numeric representation In our Q & A approach, we used Sequence To Sequence model using RNN Encoder-Decoder. It consists of two RNNs (Recurrent Neural Network) : An Encoder and a Decoder. The encoder takes a sequence(sentence) as input and processes one symbol(word) at each time step. Its objective is to convert a sequence of symbols into a fixed size feature vector that encodes only the important information in the sequence while losing the unnecessary information. You can visualize data flow in the encoder along the time axis, as the flow of local information from one end of the sequence to another.

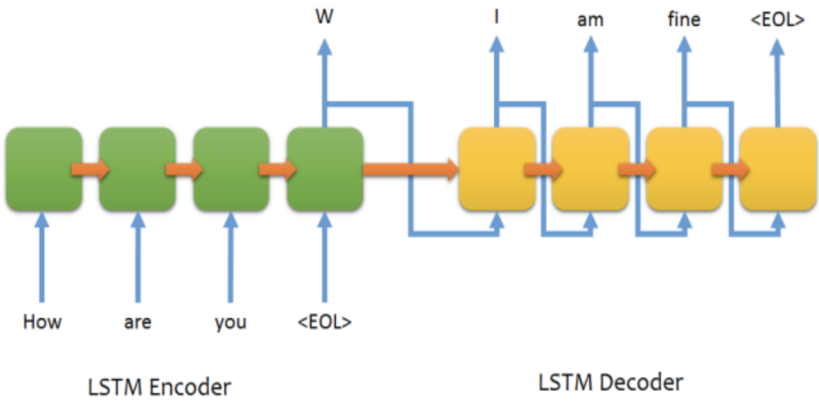


Figure 2: Simple example Sequence To Sequence model

Each hidden state influences the next hidden state and the final hidden state can be seen as the summary of the sequence. This state is called *the context or thought vector*, as it represents the intention of the sequence. From the context, the decoder generates another sequence, one symbol(word) at a time. Here, at each time step, the decoder is influenced by the context and the previously generated symbols.

References

[1] *Krishna Bhavsar, Naresh Kumar, Pratap Dangeti* Natural Language Processing with Python Cookbook // Packt Publishing, New York ,2017.pages.1-316.

[2] *Steven Bird, Ewan Klein, Edward Loper* Natural Language Processing with Python // www.nltk.org.

SOLVING PARABOLIC PDES BY MEANS OF STOCHASTIC SIMULATIONS

Oleh Pylypchak
Ukraine, Lviv
Ivan Franko National University of Lviv, Ukraine,
Faculty of Applied Mathematics and Informatics;
University of L'Aquila, Italy,
Dep. of Information Engineering, Computer Science and Mathematics
oleh.pylypchak@intermaths.eu

Introduction

Partial differential equations are in the core of modern Applied mathematics. They describe a variety of different processes in many fields of study. That's why having reliable methods for their solutions is extremely important.

We are researching an alternative method of solving PDEs, parabolic once in particular. It is a Monte Carlo method based on a link established between such equations and Brownian motion. With progress of technology, we have enough computational power to simulate this Stochastic process as many times as it takes to get an adequate approximation of its expected value. It will be shown how to use this technique to construct the solution to parabolic equation.

Problem statement

Out of all parabolic PDEs we will consider only well-posed linear parabolic PDEs[3] where, the coefficients can depend on the free variables (1 space and 1 time dimension). It can be written in the canonical form:

$$\begin{cases} -u_t + \mu u_x + \frac{1}{2}\sigma^2 u_{xx} - vu + f = 0 & x \in D,\, t \in (0, +\infty) \\ u(x,t) = \varphi(x,t) & x \in \partial D,\, t \in (0, +\infty) \\ u(x,0) = \psi(x) & x \in D, \end{cases}$$

Where, we can give physical description to each term of this equation to better understand how to proceed with the simulations:

μ – advection (drift)	σ – diffusion
v – reaction	f – forcing term (source)
φ – boundary condition	ψ – initial condition

Proposed method

Applying some of these processes to a Brownian motion[1], and using a result, that is Feynman-Kac formula[2], we can get the following representation of the solution:

$$u(x,t) = \mathbb{E}\left[\int_0^x R(r)f(X_r,r)dr + R(0)\beta(X_\tau,\tau)\ |X_t = x\right]$$

with $R(r) = e^{-\int_r^x v(X_q,q)dq}$ and $\beta(x,t) = \begin{cases} \varphi(x,t) & t > 0 \\ \psi(x) & t = 0 \end{cases}$

Therefore, our second order PDE becomes first order explicit SDE:

$$dX_t = \mu(X_t,t)dt + \sigma(X_t,t)dW_t$$

We can use some of the established numerical methods for solution of stochastic differential equation[4], simulating this process from Brownian motion W_t. After we have the values of X_t we can get an expected value of the formula above relying on the Central Limit Theorem.

Implementation

Note, that to use this method, we don't need values of the solution in any other point, so we can calculate it independently for all the points in the mesh of our domain. This fact gives us great advantages related to parallelization of the computation process and applications of such method.

Since different threads of the program do not require any communication we can use high performance computing on GPU with CUDA library. This decreases the time required for calculations more than an order of magnitude.

There are many issues to consider here. Pseudo-random numbers, that can be produced by a machine, slow down the convergence of expected value calculation. Stochastic numerical methods have 2 times lower order of convergence, then corresponding methods for ODEs. However, we can still get a good result, and such method will work in fringe cases, for example on an infinite domain (we can calculate solution on a bounded sub-mesh of it), or in a specific, interesting to us point.

References

[1] *Z. Brzeźniak* Basic Stochastic Processes // Springer, Berlin, 2002. 225 P.

[2] *R. Kohn* PDE for Finance // NYU, New York, 2015. - Lecture notes

[3] *S. Salsa* Partial Differential Equations // Springer, Milan, 2008. 556 P.

[4] *T. Sauer* Numerical Solution of Stochastic Differential Equations in Finance // GMU, Fairfax, 2010 - Lecture notes

IMAGE COMPRESSION USING MACHINE LEARNING TECHNIQUES

Yaroslav Khamar
Ukraine, Lviv
Ivan Franko National University of Lviv
Faculty of Applied Mathematics and Informatics
khamaryar@gmail.com

Digital image compression has been the focal point of a larger share of research in recent years. The aim nowadays is to not only concentrate on reducing the size of the image, but also concentrate on doing it without losing quality and information, which that image provides.

We will try to apply singular value decomposition (SVD) as one of possible machine learning image compression techniques. There are primarily two ways of compressing an image: lossless and lossy. SVD belongs to lossy image compression techniques. All the experiments will be conducted on greyscale and color images.

SVD is a significant topic in linear algebra and has many practical and theoretical values. What is special about SVD is the fact, that it can be performed on any real $m \times n$ matrices.

Let's say we have a matrix A with m rows and n columns, with rank r and $r \leq n \leq m$. Then matrix A can be factorized into three matrices as follows:

$$A = USV^T. \tag{21}$$

Here, U is an orthogonal $m \times m$ matrix,

$$U = [u_1,\ u_2,\ \ldots,\ u_r, u_{r+1},\ \ldots, u_m] \tag{22}$$

where column vectors $u_i\ \forall i = 1, 2, \ldots, m$ form an orthonormal set:

$$u_i^T u_j = \delta_{ij} = \begin{cases} 1, & \text{if } i = j, \\ 0, & \text{if } i \neq j. \end{cases} \tag{23}$$

S here is an $m \times n$ diagonal matrix with singular values on the diagonal.

Matrix V is an orthogonal $n \times n$ matrix,

$$V = [v_1,\ v_2,\ \ldots,\ v_r, v_{r+1},\ \ldots, v_n] \tag{24}$$

where column vectors $v_i\ \forall i = 1, 2, \ldots, n$ form an orthonormal set:

$$v_i^T v_j = \delta_{ij} = \begin{cases} 1, & \text{if } i = j, \\ 0, & \text{if } i \neq j. \end{cases} \tag{25}$$

As we know, every single image can be represented as a matrix of pixels. When an image is SVD transformed, it is not compressed, but rather, the data takes form in which the first singular value stores the majority of image's information. Let us illustrate the SVD image compression process [1]. Consider the following:

$$A = USV^T = \sum_{i=1}^{r} \sigma_i u_i v_i^T \tag{26}$$

With that being said, A can be represented by the outer product expansion as follows:

$$A = \sigma_1 u_1 v_1^T + \sigma_2 u_2 v_2^T + \cdots + \sigma_r u_r v_r^T \tag{27}$$

When compressing the image, the closest matrix of rank k is obtained by truncating the sums after the first k terms:

$$A_k = \sigma_1 u_1 v_1^T + \sigma_2 u_2 v_2^T + \cdots + \sigma_k u_k v_k^T \tag{28}$$

In this case, the total storage for matrix A_k will be $k(m+n+1)$. The integer k can be much less then n, and the digital image corresponding to A_k can still be very close to the original image.

In order to estimate the SVD image compression method, we can compute the compression factor as well as the quality of the obtained image. Image compression factor can be obtained using the following formula:

$$C_R = \frac{mn}{k(m+n+1)} \tag{29}$$

Here, the bigger k we take, the lower compression ratio we obtain. This is actually understandable and can be explained by the fact, that k represents the number of singular values of the matrix of the image.

In order to minimize the human factor in quality assessment of the compression of the images, we shall introduce the structural similarity index (SSIM) [2]. SSIM tries to solve the problems presented in using, e.g. the MSE method [3] by taking advantage of the human visual system. The idea of SSIM is to compute the similarity of structure between both images, rather than a pixel-by-pixel difference. SSIM gives a quality value assessment in the range of [0,1]

Having conducted several experiments, we shall analyze the obtained results from both color and greyscale images.

The results of compression of greyscale and color images of rank 256 at k with values 5, 10, 15, 20 and 25 can be seen in Table 1 and Table 2 respectively below:

k	SSIM	C_R
5	0.6611	25.55
10	0.7378	12.78
15	0.7762	8.52
20	0.8146	6.39
25	0.8392	5.11

Table 1: SVD greyscale image compression

k	SSIM	C_R
5	0.8117	42.45
10	0.8693	21.23
15	0.8890	14.17
20	0.9135	10.61
25	0.9294	8.63

Table 2: SVD RGB image compression

As for Table 1, we can see that, the bigger the number of singular values, the bigger the SSIM is. But this also means the increase in file size, which, on the other hand, can be seen through the decrease of compression ratio C_R. Also, considering steady increase in the SSIM, we see, that there is greater increase in image quality between each added singular value in the beginning than towards the end. This makes intuitive sense because singular values in the beginning are larger in magnitude than ones at the end.

From Table 2, we observe, that the general patterns have remained, but this technique requires more time and resources to execute, since for a colored image the same algorithm, applied for a single channel of a greyscale image, gets applied for each of 3 color channels of an RGB image.

References

[1] *C. J. Ogden, T. Huff* The Singular Value Decomposition and Its Applications on Image Processing // Math 45, College of the Redwoods, December 17, 1997

[2] *D. Brunet, J. Vass, E. R. Vrscay, Z. Wang* On the mathematical properties of the structural similarity index // IEEE Transactions on Image Processing. 21 (4): (April 2012) pp. 2324–2328.

[3] *B. Girod* "What's wrong with mean-squared error," in Digital Images and Human Vision // A. B. Watson, Ed. Cambridge, MA: MIT Press, 1993, pp. 207–220.

ЗАСТОСУВАННЯ СПЕКТРАЛЬНОГО МЕТОДУ СКІНЧЕННИХ ЕЛЕМЕНТІВ ДО ОДНОВИМІРНОЇ ЗАДАЧІ АДВЕКЦІЇ-ДИФУЗІЇ. АНАЛІЗ ПОХИБКИ І ПОРЯДКУ ЗБІЖНОСТІ

Соломія Гаталяк
Україна, Львів
Львівський національний університет імені Івана Франка
Факультет прикладної математики та інформатики
hatalyak.s@gmail.com

Постановка задачі

Розглянемо задачу, яка моделює процес перенесення забруднень в середовищі в одновимірному випадку.

Дану задачу описує диференціальне рівняння

$$-p\frac{d^2u}{dx^2} + Pe\frac{du}{dx} + qu = f(x), x \in (a,b) \tag{30}$$

та граничні умови

$$u(a) = 0, u(b) = 0, \tag{31}$$

де Pe - число Пекле, безрозмірна стала, p, q - сталі, $f(x)$ - задана функція.

Варіаційне формулювання

Дана задача полягає у знаходженні такої функції $u(x) \in V_0 = \{v(x) \in C[a,b], v'$-кусково неперервні та обмежені,$v(a) = 0, v(b) = 0\}$, яка задовольняє варіаційне рівняння

$$(Lu, v) = (f, v) \quad \forall v \in V_0. \tag{32}$$

Із використанням формули інтегрування частинами отримаємо

$$(Lu, v) = \int_a^b (p\frac{du}{dx}\frac{dv}{dx} + Pe\frac{du}{dx}v + quv)dx, \tag{33}$$

$$(f, v) = \int_a^b fvdx \quad \forall v \in V_0. \tag{34}$$

Спектральний метод скінченних елементів

Розіб'ємо $[a, b]$ рівновіддаленими точками на підвідрізки. На кожному з підвідрізків побудуємо базові функцій із додатковими вузлами, що розміщені в нулях сімейства ортогональних многочленів Лобатто(наприклад, 3 степеня).

Матричний запис наближеного розв'язку на підвідрізку $[x_{i-1}, x_i]$ має вигляд

$$u_i^h = N_i(x)Q_i, \tag{35}$$

де

$$N_i(x) = (\varphi_{i-1}, \varphi_{i-2/3}, \varphi_{i-1/3}, \varphi_i),$$
$$Q_i(x) = (u_{i-1}, u_{i-2/3}, u_{i-1/3}, u_i)^T$$

Із використанням (6) та варіаційного формулювання (4),(5) отримаємо вигляд матриць на підвідрізках

$$K_i = p\int_{x_{i-1}}^{x_i} (\frac{d}{dx}N_i(x))^T \frac{d}{dx}N_i(x)dx,$$

$$S_i = Pe\int_{x_{i-1}}^{x_i} (N_i(x))^T \frac{d}{dx}N_i(x)dx,$$

$$M_i = q\int_{x_{i-1}}^{x_i} (N_i(x))^T N_i(x)dx,$$

$$B_i = \int_{x_{i-1}}^{x_i} (N_i(x))^T f(x)dx,$$

$$A_i = K_i + S_i + M_i.$$

Із використанням властивостей базисних функцій сформуємо СЛАР, яку розв'язуть відомими чисельними методами.

Аналіз похибки та порядку збіжності

Простежимо за характером зміни похибок на різних сітках та числі Пекле. Знайдемо абсолютну і відносну похибки сіткового розв'язку шляхом порівняння з аналітичним розв'язком. Для цього використаємо норми в просторах L_2 та $W_2^{(1)}$. При обчисленні похибок у нормі $W_2^{(1)}$ похідні від чисельного розв'язку усереднюються у вузлових точках.

Для оцінки порядку збіжності скористаємось співвідношенням

$$p = \frac{ln\delta_h - ln\delta_{\frac{h}{2}}}{ln2},$$

де δ_h - абсолютна похибка при кроці дискретизації h у відповідному просторі.

Література

[1] *Савула Я. Г.* Числовий аналіз задач математичної фізики варіаційними методами. // Видавничий центр ЛНУ імені Івана Франка, Львів, 2004. ст.30-37.

[2] *C. Pozrikidis.* Introduction to Finite and Spectral Element Methods Using MATLAB, Second Edition. // CRC Press, Boca Raton, 2014. pages.154-158,165-174.

[3] *Patrick Knupp, Kambiz Salari* Verification of computer codes in computational science and engineering. // Chapman and Hall/CRC, Boca Raton, 2003. pages.24-30.

[4] *Jan Pech* Application of spectral element method in fluid dynamics. // Institute of Theoretical Physics, Czech Republic, 2013. pages.20-23.

ПОБУДОВА ВЕБ-ЗАСТОСУНКУ ДЛЯ РОЗПІЗНАВАННЯ ТА ТРИАНГУЛЯЦІЇ ШКІРНИХ ОБЛАСТЕЙ ЗА ДОПОМОГОЮ ТЕХНОЛОГІЙ NODE.JS ТА VUE.JS

Корольчук Ярослав, Кондор Марта, Мудрик Андрій
Україна, Львів
Львівський національний університет імені Івана Франка
Факультет прикладної математики та інформатики
koroltchukjaroslav@gmail.com, martakondor17@gmail.com,
mudrykandrew@gmail.com

Вступ

У наш час веб-сторінки виконують найрізноманітніші функції, від джерела інформації до миттєвої комунікації між людьми. Головною їх перевагою є доступність, адже для того щоб використати веб-застосунок достатньо мати будь який пристрій з доступом до мережі інтернет.

Актуальність даної роботи полягає в тому, що Метод скінченних елементів(МСЕ) для знаходження розв'язків інтегральних та диференціальних рівнянь у частинних похідних набуває популярності, адже розроблений для чисельного розв'язування складних проблем у механіці конструкцій та інших математичних задач. Наразі широко застосовується для розв'язування складних систем. Як наслідок постає питання автоматизації роботи, що спростить і пришвидшить обчислення.

Першим етапом МСЕ є виділення області та розбиття її на підобласті. Саме таку функціональність було вирішено покласти як основу у дану веб-аплікацію. Отже, Мета дослідження полягає в розробці веб-додатку для автоматизації пошуку та тріангуляції областей зручних для подальшого застосування МСЕ.

Головна ідея

Процес розробки керувався таким правилом: веб-застосунок повинен бути зручний та простий у застосуванні будь-яким користувачам. Тому все що потрібно користувачеві це додати фотографію та натиснути кнопку тріангуляції.

Технічно все можна уявити як на малюнку нижче. Користувач може бачити тільки зовнішній рівень веб-аплікації позначений зеленим кольором. Тут використовується Vue.Js як сучасний JavaScript фреймворк для розробки аплікацій, а також Bulma – CSS фреймворк для

зручного та приємного дизайну веб-аплікації. Цей рівень комунікує із Node.Js сервером, який є керівником операцій аплікації та виконує дві основні функції:

- Перевіряє доступ випадкових користувачів до функціональності аплікації
- Зберігає фотографію та інформує Python сервер про необхідність тріангуляції

База даних виконує місце сховища інформації як про юзерів, що мають доступ до аплікації, так і до місцезнаходження фотографій юзерів

Сам процес розпізнавання відбувається на сервері Python. Після отримання фотографії, ми забираємо шум з зображення і згладжуємо його для легшого розпізнавання. Далі машинним навчанням розділяємо область на підобласті (шкіру і захворювання). Отримуємо контури захворювання і апроксимуємо їх для того щоб зменшити кількість точок у контурі. Після чого проводимо триангуляцію по вихідних точках і повертаємо вже триангульовану область.

Аплікація могла бути реалізована і без використання веб-застосування, але тоді користувачу довелось б вивчати специфіку використання REST арі серверу. Це загально прийняті принципи побудови веб-застосунків, де вся робота з даними відбувається на захищеному сервері, що робить користування аплікацією:

- Захищеним (При логуванні користувачів з базою звіряються не основні паролі, які можна б було легко отримати із доступом до бази даних, а хешована версія яку практично неможливо отримати, якщо не пройти через етап хешування на Node.Js сервері)
- Швидким і простим (Навантаження опрацювання даних іде на сервер, а не на клієнт користувача, тому користування аплікацією можливе на мінімальних системних вимогах)

На основі вище зазначених плюсів веб-застосунку ми переконались у правильності вибору веб-сторінки як графічного інтерфейсу нашої функціональної частини.

Висновок

Веб-застосунки у сучасних інформаційних технологіях є незамінними джерелами інформації, адже полегшують роботу з даними та одночасно дозволяють працювати з інформацією в захищеному режимі.

Підсумовуючи усе вище сказане, можемо зробити висновок, що наша веб-аплікація може в подальшому використовуватись для генерації базових сіток для МСЕ, адже вона дозволить користувачам у декілька натискань отримати протриангульовану область з вхідної фотографії.

Література

[1] *Савула Я. Г.* Числовий аналіз задач математичної фізики варіаційними методами. // Видавничий центр ЛНУ імені Івана Франка, Львів, 2004. ст.30-37.

[2] https://nuxtjs.org/

[3] https://bulma.io/

МОДИФІКАЦІЯ ТРИКОВОГО ГРАДІЄНТНОГО МЕТОДУ МІНІМІЗАЦІЇ ФУНКЦІЇ

Герилів Назар Богданович, Україна
ЛНУ імені Івана Франка, факультет прикладної математики та інформатики
e-mail: nazarheryliv12@gmail.com

Розглянемо задачу безумовної мінімізації $f\left(x\right) \to inf, \quad x\epsilon R^n$ (1)

Будь-який числовий метод розв'язування задачі оптимізації ґрунтується на точному або наближеному обчисленні певних характеристик(значення функції мети, значення похідних тощо). На основі отриманої інформації будуємо наближення до розв'язку задачі. Для кожної конкретної задачі питання про характеристики функції, які потрібно брати до уваги під час розв'язування задачі, залежить від властивостей функції.
Для мінімізації функцій багатьох змінних використовуємо послідовні алгоритми, коли вибір наступного наближення залежить від попередніх результатів у цьому випадку застосовуємо алгоритм вигляду.

$$x_{k+1} = x_k + \alpha_k h_k, \qquad \text{де } \alpha_k \in R, \quad h_k \in R^n, \quad k = 0,\ 1, \ldots$$

Розглянемо простий градієнтний метод в якому $h_k = -f'(x_k)$, отже послідовність визначається за формулою

$$x_{k+1} = x_k - \alpha_k f'(x_k), \qquad \alpha_k > \ 0, \quad k = \ 0, 1, \ldots$$

Оскільки реальний обчислювальний процес повинен бути скінченним, отже, для відповідного методу повинні існувати умови зупинки. Найчастіше на практиці використовують такі:

$$||x_{k+1} - x_k|| \leq \varepsilon_1$$

$$||f(x_{k+1}) - f(x_k)|| \leq \varepsilon_2$$

$$||f'(x_{k+1})|| \leq \varepsilon_3$$

Використавши схему побудови трикрокових ітераціних методів утворюємо метод, який має вигляд:

$$v_k = x_k - \alpha_k f'(x_k),$$

$$u_k = \ x_k - \ \gamma_k \left(\lambda f'\left(x_k\right) + (1 - \ \lambda)\, f'\left(v_k\right)\right), \ \lambda\epsilon\left(0, 1\right),$$

$$x_{k+1} = u_k - \beta_k(u_k - v_k),$$

де

$$f(u_k - \ \beta_k(u_k - \ v_k)) = \min_{\beta}(u_k - \beta(u_k - \ v_k)).$$

Поруч з запропонованими алгоритмами, розглянуто деякі модифікації:
Для цього введемо такі позначення

$$\mathrm{N} = \{1, 2, \ldots, \mathrm{n}\}, \quad N_i = \{i_1, \ldots, \ i_s\}, \ \text{ де } 1 \le i_j \le n,$$

$$i_j \ \neq \ i_v, \ i = 1, 2, \ldots, k \le n, \ \ \cup N_i = N, \ \ N_i \ \neq \ N_j$$

Розглянемо вектори

$$h_i = (h_i^1, \ldots, h_i^n)^T \text{ де } \ h_i^j(x) = \begin{cases} 0, & i \notin \ N_i \\ \frac{\partial f(x)}{\partial x^i}, & i \in N_i \end{cases}$$

Легко показати виконання умови $(h_i, f'(x)) > 0$, тобто вектор -h_i є вектором спадання функції $f(x)$ у точці x. Крім того виконується умова $(h_i, h_j) \ge 0$, тобто -h_i, -h_j вектори спадання функції $f(x)$ у точці x, причому кут між векторами h_i, h_j не тупий.
Тепер розглянемо алгоритм $x_{k+1} = x_k - \ \alpha_k h_{-\left[\frac{k}{s}\right]s+k+1} \qquad k = 0, 1, \ \ldots$
При виконанні умов послідовність $\{x_k\}$ збігається до точки x_*, яка є стаціонарною точкою функції $f(x)$. У випадку $N_i = \{i\}$ ми матимемо покоординатний метод спуску. У випадку $s_i = \ s_j > 1$ матимемо блочний метод. Враховуючи цей факт можна побудувати трикрокові алгоритми розв'язування задачі (1), а саме

$$u_k = x_k - \alpha_k h_{-\left[\frac{k}{s}\right]s+k+1}$$

$$v_{k+1} = x_k - \beta_k h_{-\left[\frac{k+1}{s}\right]s+k+2}$$

$$x_{k+1} = \operatorname*{argmin}_{\gamma} f(\ u_k + \gamma(v_k - \ u_k)). \quad k = 0, \ 1, \ \ldots$$

Даний алгоритм є по своїй структурі у випадках великих n і відповідному виборі h_i менш трудомісткий ніж класичний, градієнтний метод в сенсі кількості обчислень.

1. *М.Я.Бартіш* Методи оптимізації. Теорія і алгоритми. Львів Видавничий центр ЛНУ імені Івана Франка, 2006, - 222 с.

2. https://studfiles.net/preview/5401509/page:2/

3. https://uk.wikipedia.org/wiki/

КЛАСТЕРИЗАЦІЯ ВИПАДКОВИХ ДАНИХ

Степаняк О.Б., Чабанюк Я.М.

Львів, Україна

ЛНУ ім. Івана Франка

Факультет прикладної математики та інформатики

oleg.stepaniak.01.07.1998@gmail.com

Побудова кластерів на множинні випадкових об'єктів. За допомогою модифікацій ітераційних алгоритмів K-Mean [1,3] та FOREL [2,4,5] здійснено побудову кластерів. При цьому закладено випадковий характер початкового наближення кластерів. Модифікація вказаних алгоритмів полягає в обробці нових даних, що теж мають випадкову природу.

Зокрема, у методі K-Mean задається випадкове початкове наближення та випадкова кількість кластерів. Такі умови в об'єднанні з полем випадкових об'єктів гарантують різні результати роботи алгоритму. У алгоритмі FOREL випадково згенеровано центр гіперсфери, що змінить кількість та обсяг кластерів.

Проведено дослідження недоліку методу K-Mean , а саме його не збіжність [6]. Запропоновано варіанти виправлення даної проблеми.

ЛІТЕРАТУРА

1. *Chandan K. Reddy ,Charu C. Aggarwal,* "Data Clustering: Algorithms and Applications" //CBC Press, 2013.pages.532.

2. *Rui Xu, Donald C. Wunsch,* "Clustering", 2009.pages.358.

3. "What is Cluster Analysis?"

http://www.stat.columbia.edu/ madigan/W2025/notes/clustering.pdf

4. *Tariq Rashid,Osmar R. Zaïane,* "A Tutorial on Clustering Algorithms"

,

http://www.stat.columbia.edu/

5. "Clustering in Machine Learning" ,

https://www.geeksforgeeks.org/clustering-in-machine-learning/par 6. *Andrea Trevino,* "Introduction to K-Mean clustering" ,

https://www.datascience.com/blog/k-means-clustering

NON LINEAR MIXED EFFECT MODEL BASED APPROACH ON THE PHARMACOKINETICS-PHARMACODYNAMICS OF REMIFENTANIL

Damilola Victoria Tomori
Ukraine, Lviv
Ivan Franko National University of Lviv
Faculty of Applied Mathematics and Informatics
damilola.tomori@intermaths.eu

A move towards the adoption of population pharmacokinetics (PK) as a routine procedure during drug development should now be encouraged. A number of studies have shown that it is possible to organise existing, routine data in such a way that valuable information on pharmacokinetic-pharmacodynamics (PKPD) variability can be obtained. In the last decade, population pharmacokinetic (PopPK) modeling has spread its influence in the whole process of drug research and development. This change in emphasis has been largely brought about by the development of appropriate computer software (NONMEM: Nonlinear Mixed Effects Model) and its successful application to the retrospective analysis of clinical data of a number of commonly used drugs, e.g. digoxin, phenytoin, gentamicin, remifentanil, procainamide, mexiletine and lignocaine . While targeting the construction of the dose-concentration of a drug based on a population of patients, it is an efficient means of studying several factors and shows great flexibility in dealing with sparse or rich samplings.

Firstly, we aimed to establish the place of PopPK-PD modelling, using non linear mixed effects models to obtain PK-PD information in drug development in sixty-five healthy adults ages 20 to 85 years by constant rate infusion of 1 to 8 $\mu g^{-}kg^{-1}$. Secondly, we statistically evaluate the dynamics of the concentration administered to the subjects by using the relationships established between model parameters to explain PKPD variability and facilitate dose adjustment decisions. The data used in this study originally published in 1997 by American Society of Anesthesiologists where the study identified the influence of age and gender on the PKPD of remifentanil by the used of generalized additive model implemented in NONMEM. MONOLIX, an advanced and simple solution for non-linear mixed effects modeling which is based on the Stochastic Approximation Expectation-Maximization(SAEM) in conjuction with R programming language will be used for the analysis of this work.

First step required in the development of PopPk model is to identify the base or structural model, which is the model that best describes the data in the absence of covariates. The observed concentration-time profiles of remifentanil were best described by a three-compartment model with a linear elimination rate via an intraveneous infusion. PK three-compartment model divided the body into central compartment and two peripheral compartments. The central compartment (compartment 1) consists of the plasma and tissues where the distribution of the drug is practically instantaneous. The peripheral compartments (compartments 2 and 3) consist of tissues where the distribution of the drug is slower compared to compartment 1. Drug concentrations in the central and peripheral compartments is thus described with differential equations:

$$\begin{cases} \dot{C_1}(t) = v - (k_{12} + k_{13} + k)C_1(t) + k_{21}C_2(t) + k_{31}C_3(t) \\ \dot{C_2}(t) = k_{12}C_1(t) - k_{21}C_2(t) \\ \dot{C_3}(t) = k_{13}C_1(t) - k_{31}C_3(t) \end{cases} \tag{36}$$

$$\textit{with initial conditions} \begin{cases} C_1(t) = 0 \quad \text{for } t = t_D \\ C_2(t) = 0 \quad \text{for } t = t_D \\ C_3(t) = 0 \quad \text{for } t = t_D \\ C_1(t_D) = \frac{D}{v} \end{cases}$$

The corresponding solution to equation (1) is giving as

$$A = \frac{1}{v}\frac{K_{21} - \alpha}{\alpha - \beta}\frac{K_{31} - \alpha}{\alpha - \gamma},\ B = \frac{1}{v}\frac{K_{21} - \beta}{\beta - \alpha}\frac{K_{31} - \beta}{\beta - \gamma},\ C = \frac{1}{v}\frac{K_{21} - \gamma}{\gamma - \beta}\frac{K_{31} - \gamma}{\gamma - \alpha}$$

$$C(t) = \begin{cases} \frac{D}{Tinf}[\frac{A}{\alpha}(1 - \exp -\alpha(t - t_D) \\ +\frac{B}{\beta}(1 - \exp -\beta(t - t_D) \\ +\frac{C}{\gamma}(1 - \exp -\gamma(t - t_D)], & \text{if } (t - t_D) \leq Tinf \\ \\ \frac{D}{Tinf}\frac{A}{\alpha}(1 - \exp \alpha Tinf)(\exp -\alpha(t - t_D - Tinf)) \\ +\frac{B}{\beta}(1 - \exp \beta Tinf)(\exp -\beta(t - t_D - Tinf)) \\ +\frac{C}{\gamma}(1 - \exp \gamma Tinf)(\exp -\gamma(t - t_D - Tinf)), & \text{otherwise} \end{cases} \tag{37}$$

The equations express the concentration C(t) in the central compartment at a time t after the last drug administration, D is the total dose administered, the duration of infusion is $Tinf$, v is the volume of distribution, k is the elimination rate constant, α, β and γ are the first rate constant, second rate constant and third rate constant respectively. K_{12}, K_{21} are

the distribution rate constant from compartment 1 to compartment 2 and from compartment 2 to compartment 1 respectively. K_{13}, K_{31} are distribution rate constant from compartment 1 to 3 and from compartment 3 to 1 respectively and c_0 is the initial drug intake.

Once the base model is identified, covariate sub-models are developed. We introduced age, lean body mass, gender and infusion categorical rate to the structural model. A complete PopPK model consists of a structural model, statistical model and a covariate model where the choice of the structural model has implications for covariate selection. Approximately, thirty different covariates submodels were considered. Bayesian information criterion (BIC) value obtained for the based model (no covariates) is 7438.66. After which, covariates was introduced, where a model with the highest covariates (10 covariates) was selected by BIC (7347.41) as the final covariate model with an inclusion of LBM on all parameters with AGE on k_{21} , AGE on k_{13}, AGE on v, AGE on k. To improve the final covariate model, we considered correlations between the random effects , a Wald test which strongly relies on the standard error was performed and it thus suggest to remove k_{13} and k_{31} from LBM with the corresponding β coefficients, $\beta_{k_{31}LBM}$, $\beta_{k_{13}LBM}$ with p values of 0.991 and 0.614 respectively. The model BIC value is now 7113.02, which is an improvement to the final model.

References

[1] *Whiting B, Kelman AW, Grevel J.* Population Pharmacokinetics. Theory and clinical application Pubmed, United state ,1986.

[2] *Charles F, Minto,M.B., Ch.B., Thomas W. Schnider, M.D., Talmage D. Egan, M.D.* Influence of Age and Gender on the Pharmacokinetics and Pharmacodynamics of Remifentanil ,American society of Anaestheologist,Lippincott-Raven publisher City ,1997.

[3] *Marc Lavielle* mixed Effect Model for the population Approach, Models, Tasks, Methods and Tools , CRC press, Newyork ,2015.

НЕЙРОННІ МЕРЕЖІ ДЛЯ РОЗПІЗНАВАННЯ РУКОПИСНОГО ТЕКСТУ

Баранов Микола
Україна, Львів
Abto Software
office@abtosoftware.com

Задача автоматичного розпізнавання тексту на зображенні має багато сфер практичного застосування: розпізнавання номерних знаків машин, отримання інформації про продукт, оцифрування рукописних документів тощо. Оскільки цю задачу важко вирішити методами традиційного програмування, на допомогу приходять нейронні мережі

Загальна постановка задачі роботи із зображенням

Зображення можна розглядати як матрицю чисел, де кожен елемент матриці репрезентує відповідний піксель зображення з заданою інтенсивністю (кольором). Таке подання зображення дає змогу представити його у вигляді дискретної двовимірної функції

$$I(x, y)$$

Задання зображення шляхом функції відкриває широкі можливості використання математичного апарату. Проте, очевидно, що складні задачі з області комп'ютерного бачення важко розв'язати прямими алгоритмами. Це завдання можна вирішувати нейронними мережами

Концепція нейронних мереж

Штучні нейронні мережі покликані знаходити залежності між вхідними та вихідними даними. Такі типи залежостей вдається встановити завдяки статистичному аналізі даних. Очевидно, що більша кількість даних забезпечить кращий результат. В загальному, вигляд нейронної мережі можна звести до математичної функції

$$f(x, \overline{w}) = y$$

де x - вхідні данні, y - результат роботи мережі, $\overline{w}$ - внутрішні параметри (ваги) мережі. Процес тренування полягає у автоматичному пошуку усіх параметрів $w_i \in \overline{w}$. Для тренування необхідно підготувати набір даних з очікуваними виходами нейронної мережі

$$\overline{X} = \{x_{0,}x_1,..., x_n\}, \overline{Y} \in \{y_{0,}y_1,..., y_n\}$$

Використовуючи оптимізаційні методи ми можемо відшукати найкращі значення параметрів $w_i \in \overline{w}$
Параметри x, y є сталими, $w_i \in \overline{w}$ відшукуються під час тренування. Потрібно знайти щагальний вигляд функції f. Існує багато архітектур нейронних мереж, але більшість з них складаються з базових частин — шарів.

$$f(x, \overline{w}) = l_n(l_{n-1}(...l_2(l_1(x, \overline{x_1}), \overline{w_2})...\overline{w_{n-1}}), \overline{w_n})$$

$$\overline{w} = \overline{w_1} \cup \overline{w_1} \cup ... \cup \overline{w_b}$$

У найпростішій щільно з'єднаній мережі (fully conected) базовим блоком є однойменни шар, який виконує множення вектора x на матрицю $\overline{w}$ та застовує нелінійну функцію активації. Проте, цей тип мереж не є найкращим для обробки зображень.

Згорткові нейронні мережі (CNN)

У реальних зображення легко прослідковується кореляція між сусідніми пікселями. Відповідно до теорії інформації, чим більша статистична залежність даних — тим меншу інформацію ці дані несуть. Згорткові шари водночас опрацьовують певну локальну облать зображення та не враховують кореляції з далекими частинами, використовують математичну операцію згортки.

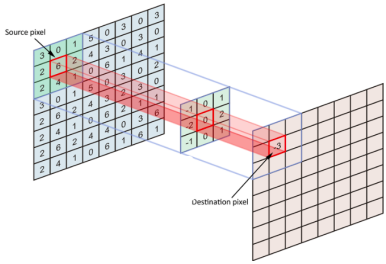


Рис. 1: Дискретна операція згортки

$$(f * g)(t) = \int_{-\infty}^{\infty} (f(\tau)g(t - \tau))\, d\tau$$

Використання цієї технології в поєднані з щільними шарами нейронних мереж дає потужний апарт для класифікації, сегментації зображень тощо.

Рекурентні шари (RNN)

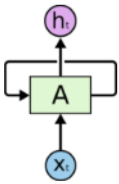


Рис. 2: Рекурентний шар

Описані вище шари нейронних мереж працюють як строга математична функція. Одинакові вхідні данні спричинять завжди однаковий результат. Для певного класу задач необхідно пам'ятати попередні події (наприклад, задача розпізнавання голосового тексту). Рекурентні шари пропонують додаткові акумулятори пам'яті всередині шару. Кожен результат мережі подається додатковим вхідним параметром в наступний момент часу. Звідси і назва — рекурентні нейронні мережі.

Згортково-рекурентні мережі для розпізнавання тексту (CRNN)

Завдання розпізнавання тексту можна звести до завдання розпізнавання окремого слова, оскільки сегментація слів на зображенні — задача підвладна традиційним алгоритмам комп'ютерного бачення. Згорткова частина мережі використовується для витягування особливостей локальних частин зображення. Розумно очікувати, що отримана інформація буде придатною для класифікації окремих літер. Проте, дана класифікація не є стійкою, оскільки ймовірність правильно розпізнати слово є досить малою

$$p = p_1 \cdot p_2 \cdot .. \cdot p_n$$

і складається з добутку ймовірностей правильної класифікації літер. Але ймовірність наступного символу беспосередньо залежить від попередніх символів. Рекурентні шари в процесі навчання успішно вивчають загальні правила написання слів та правильно "читають" навіть помилково написану літеру.

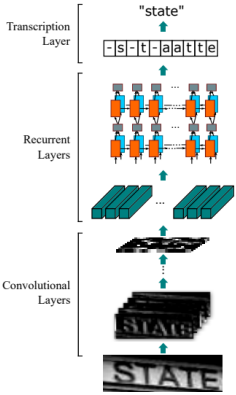
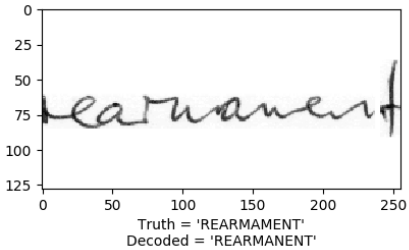
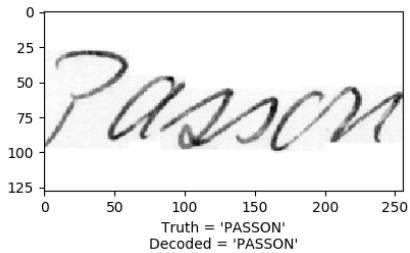


Рис. 3: Вигляд CRNN

Для експерименту було обрану навчальну базу англійських слів IAM Handwriting database, 115 тисяч зображень слів. *Отримані результати:*

- Точність: 75%
- Середня edit distance: 0.1





Література

[1] An End-to-End Trainable Neural Network for Image-based Sequence Recognition and Its Application to Scene Text Recognition /Baoguang Shi, Xiang Bai and Cong Yao - arXiV, 2015

[2] Neural Networks for Pattern Recognition / M. Bishop - Oxford, 1999

[3] Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) Network / Alex Sherstinsky - arXiV 2018

[4] Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling /Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, Yoshua Bengio - arXiV, 2014

[5] Deep Learning with TensorFlow: Explore neural networks with Python /Giancarlo Zaccone, Rezaul Karim - Paperback, 2017

ТОЧНІ РОЗВ'ЯЗКИ НЕЛІНІЙНОГО РІВНЯННЯ В ЧАСТКОВИХ ПОХІДНИХ МЕТОДОМ (G'/G)-РОЗШИРЕННЯ

Кравчук Ольга
Україна, Львів
Львівський національний університет імені Івана Франка
Факультет прикладної математики та інформатики
germiona18@gmail.com

Розглянемо КдФ рівння Бюргерса[1]:

$$u_t + puu_x + qu^2u_x + ru_{xx} - su_{xxx} = 0, qs > 0 \quad (1)$$

Знайдемо його розв'язок методом (G'/G)-розширення[2].
Дане рівняння досліджуємо використовуючи таку послідовність кроків.
Крок 1. Введемо змінну ξ та виконаємо таку заміну:

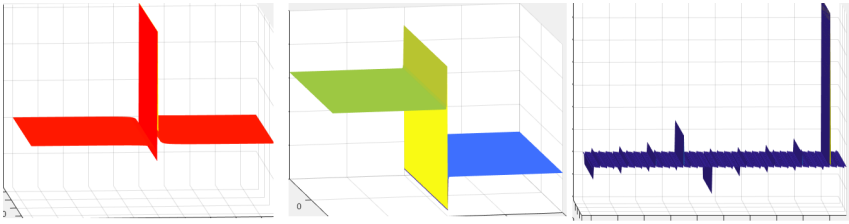
$$u(x,t) = u(\xi), \xi = x - Vt,$$

де V – константа. Внаслідок даної заміни отримуємо звичайне диференціальне рівняння
$-Vu' + puu' + qu^2u' + rx - su''' = 0.$ (2)
Крок 2. Розв'язки рівняння (2) шукаємо у вигляді
$\sum_{i=0}^{m} a_i(\frac{G'}{G})^i, a_m \neq 0.$ (3)
Для знаходження m розглянуто однорідний баланс між u^2u' і u'''. Отримано m = 1. Отже розв'язок рівняння (1) буде мати вигляд
$u(\xi) = a_1\left(\frac{G'}{G}\right) + a_0, a_1 \neq 0$ (4)
де $\left(\frac{G'}{G}\right)$ задовольняє звичайне диференціальне рівняння
$G'' + \lambda G' + \mu G = 0$ (5)
Крок 3. Підставляючи (4) у (2) та звівши члени біля однакових степенів $\left(\frac{G'}{G}\right)$, отримуємо многочлен від $\left(\frac{G'}{G}\right)$ у лівій частині рівності і прирівнюємо кожен коефіцієнт цього многочлена до нуля. В результаті отримаємо систему алгебричних рівнянь щодо параметрів
$a_0, a_1, \mu, \lambda, V, p, q, s, r.$
Крок 4. Розв'язавши систему з попереднього кроку отримаємо
$a_0, a_1, \mu, \lambda, V, p, q, s, r.$ Розглянемо один із отриманих розв'язків:

$$v = 0, \ \mu = \frac{-3sp^2 + 2qr^2}{24qs^2}, \ \lambda = 0,$$

$$a_0 = -\frac{p}{q} - \frac{r\sqrt{\frac{s}{q}}}{\sqrt{6}s} \quad a_1 = -\sqrt{\frac{6s}{q}}$$

де p,q,s,r – довільні константи.
Підставляємо розв'язки алгебричної системи у (4) та отримуємо гіперболічний, тригонометричний та раціональний розв'язки.
Графіки раціонального, гіперболічного та тригонометричного розв'язків:



Література

[1] *E. J. Parkes, B. R. Duffy* Travelling solitary wave solutions to a compound KdV-Burgers equation / Department of Mathematics, University of Strathclyde, Glasgow, 1997

[2] *M. Wang, X. Li, J. Zhang* The (G'/G)-expansion method and travelling wave solutions of nonlinear evolution equations in mathematical physics / Phys. Lett. A, №372, 2008, c. 372–417.

ВЕБ-ОРІЄНТОВАНИЙ ДОДАТОК ДЛЯ ВІДСТЕЖЕННЯ ЛАБОРАТОРНИХ РОБІТ TASKER

Гайдуков Ростислав В'ячеславович
Україна, Чернівецький національний університет імені Юрія Федьковича, Факультет математики та інформатики
rostyslav.haidukov@live.com

В епоху інформаційних технологій процес слідкування викладача за виконанням студентами лабораторних робіт вимагає автоматизації.

У зв'язку із активним розвитком веб-технологій, великої популярності набувають веб-додатки, що надають досвід користувача близький до користування звичайними Desktop-додатками. Все більше додатків тепер можуть похизуватися веб-версіями, в той же час існуючі веб-додатки переносять більше логіки на клієнтську сторону — на Front-End.

Було вирішено розробити такий додаток, так званий Single-Page application. Додаток дозволяє викладачу створити певний робочий простір. Викладач може за посиланням запросити студентів. У кожному робочому просторі можна створити спринт, і туди додавати варіанти лабораторних робіт. Студент може бачити перелік робочих областей, куди його запросили. Студент має свою дошку, де він може бачити стовпці із групами завдань (Todo, In Progress, Done). Студент може перетягнути завдання із спринта з колонки "Todo"в "In Progress". Викладач має повний доступ до дошки коржного студента. Після виконання завдання його треба перетягнути в "Done".

При розробці Single-Page додатків доводиться монолітний додаток розбивати на 2 окремі додатки: клієнтський і серверний. Спілкуються вони між собою по REST. Для збереження даних на клієнтській стороні використано WebStorage API.

Для розробки Front-End було використано широко відому бібліотеку React [1]. Вона дозволяє декларативно описувати складні інтерфейси веб та мобільних додатків. За рахунок використання технології Virtual DOM при будь-яких змінах в інтерфейсі перерендуватися буде тільки те, що змінилося, а не ціла сторінка.

В якості СКБД обрано нереляційну документоорієнтовану СКБД MongoDB [3]. Вона класифікується як NOSQL, використовує JSON-подібні документи для схеми даних. Основною перевагою є простота та гнучкість JSON формату, відсутність жорстко заданої схеми, що призводить до високої швидкості розробки.

Серверна частина (Back-End) реалізована на Node.js [2]. Це середовище JavaScript, яке надає неблокуючу асинхронну модель для

операцій вводу-виводу.

Література

[1] *Алекс Бэнкс, Ева Порселло.* React и Redux: функциональная веб-разработка. СПб.: Питер, 2018. — 336 p.

[2] *Mike Cantelon, Mark Harter, TJ Holowaychuk, Nathan Rajilich.* Node.js in Action. — NY: Manning Publications Co., 2014. — 396 p.

[3] *Kyle Banker, Peter Bakkum, Shaun Verch, Douglas Garrett, Tim Hawkins.* MongoDB in Action. — NY: Manning Publications Co., 2016. — 480 p.

ДЕЦЕНТРАЛІЗОВАНІ ДОДАТКИ НА ОСНОВІ БЛОКЧЕЙН

Кирило Возний, Сергій Проц, Денис Антош
Україна, Чернівці
Чернівецький національний університет імені Юрія Федьковича
Факультет математики та інформатики
k.vozniy@gmail.com, prots.sergiy7@gmail.com, anton.denis98@gmail.com

В епоху цифрових технологій децентралізованість даних грає велику роль в захисті інформації від підробок, її цілісності. Саме тут у гру вступає блокчейн, технологія, що набула великої популярності в останні роки. Воно й не дивно, адже завдяки ньому ми можемо безпечно зберігати свої дані, не переживаючи за їх цілісність, розпоряджатись своїми фінансами, виконувати важкі обчислення на декількох машинах одночасно й багато іншого.

Блокчейн дослівно означає «ланцюжок блоків». Це база даних, яка зберігається одночасно на безлічі комп'ютерів. Нові блоки в цій базі-ланцюжку створюються постійно. Кожен новостворений блок містить групу впорядкованих записів (транзакцій), що накопичилися за останній час, а також заголовок. Коли блок сформований, він перевіряється іншими учасниками мережі і потім, якщо всі згодні, під'єднується до кінця ланцюжка. Як тільки це сталося, внести в нього зміни вже неможливо. Крім нової інформації блок також зберігає в зашифрованому вигляді дані про попередні блоки. А база автоматично оновлюється на всіх підключених до системи пристроях.

Одна з найпопулярніших платформ, що створена на базі блокчейну - Етеріум [1]. Це платформа для створення практично будь-яких децентралізованих онлайн-сервісів на базі блокчейна (Dapps), що працюють на базі розумних контрактів. Реалізована як єдина децентралізована віртуальна машина. Ідея була втілена 30 липня 2015 року. Оскільки Ethereum сильно спрощує і здешевлює впровадження блокчейна, його впроваджують як великі гравці, такі як Microsoft, IBM, Acronis, Сбербанк, банківський консорціум R3, так і нові стартапи.

Для написання смарт-контрактів на базі Етеріум використовується об'єктно-орієнтована та предметно-орієнтована мова програмування Solidity [2]. Це статично типізована JavaScript-подібна мова програмування, створена для розробки розумних контрактів, які працюють на віртуальній машині Ethereum (EVM). Програми на мові Solidity транслюються в байткод EVM. Solidity дозволяє розробникам створювати самодостатні програми, що містять бізнес-логіку, результуючу в транзакційні записи блокчейну. Підтримуються комплексні змінні

контрактів, включаючи довільні ієрархічні відображення (mappings) і структури. Контракти підтримують спадкування, включаючи множинне і C3-лінеаризацію. Підтримується бінарний інтерфейс програмування (ABI), що має безліч типобезпечних функцій в кожному контракті.

Для взаємодії зі Ethereum нодами використовуючи HTTP або IPC з'єднання була задіяна колекція бібліотек web3.j [3]. Завдяки ній ми можемо отримувати інформацію з децентралізованої бази даних, розпоряджатись власними токенами, надсилати інформацію на ноди, тощо.

MetaMask - це розширення для доступу до розподілених додатків Ethereum, або "Dapps"у браузері. Розширення вбудовує API Web2 Ethereum у контекст JavaScript кожного веб-сайту, так що dapps можуть читати з блокчейн мережі. MetaMask дозволяє користувачеві створювати і керувати власними ідентифікаціями, тому, коли децентралізований додаток хоче виконати транзакцію і записати в блокчейн, користувач має змогу затвердити або відхилити її.

Під час розробки було створено додаток, за допомогою якого серед всієї групи користувачів обирається керівник методом голосування. Кожний з користувачів спочатку має приєднатися до голосування. Далі він може подати свою кандидатуру та проголосувати за себе або за іншого кандидата. Зареєстрований в голосуванні користувач може бачити хто приєднався до голосування, хто подав кандидатуру та хто вже проголосував. В кінці він може подивитись чи всі зареєстровані користувачі проголосували та побачити результат цього голосування.

Особливістю платформи Ethereum є доволі легка методика створення власних токенів. Найпростішим є стандарт ERC-20 [4], що визначає набір правил, які повинні бути дотримані для того, щоб токен був прийнятий і мав можливість взаємодіяти з іншими токенами в мережі. Самі токени є блокчейн-активи, які можуть мати цінність, а також можуть бути відправлені й отримані як будь-яка інша криптовалюта.

Також було створено додаток, що дає змогу ближче ознайомитися з розробкою власного токену та розглянути сферу використання токена як валюти. А саме додаток, що дозволяє користувачам брати участь в лотереї, переглядати попередні транзакції та ігри, використовуючи технологію блокчейн та алгоритм Keccak для знаходження випадкового числа, розпоряджаючись власними токенами, а також мати змогу конвертації валюти Ethreum в токени та навпаки.

Для відображення інформації було використано javascript фреймворк Vue.js [5]. Vue використовує синтаксис шаблонів на основі HTML, що дозволяє декларативно зв'язувати рендеринг DOM з основними

екземплярами даних в Vue. Всі Vue шаблони валідні HTML, і можуть бути розпарсені браузерами та HTML парсерами. Всередині Vue компілює шаблони в рендерингові функції віртуального DOM. В поєднанні з реактивною системою, Vue здатний розумно обчислити кількість компонентів для ре-рендингу та застосувати мінімальну кількість маніпуляцій з DOM, коли стан застосунку зміниться. Для стилізації компонентів на сторонці використовував Vue-Bootstrap - бібліотеку яка надає можливість, швидко й просто користуватись готовими, стилізованими візуальними об'єктами.

У процесі розробки був створений ще один додаток, що дозволяє користувачам проводити аукціон з продажу майна, використовуючи технологію блокчейн та розпоряджатись власними токенами. Користувач має змогу виставити майно на продаж, вказавши його мінімальну ціну, час впродовж якого його можна буде придбати, за потреби відмінити замовлення про продаж. В свою чергу потенційний покупець може запропонувати власну ціну за товар, який його зацікавив або купити товар за фіксованою вартістю.

В результаті було розроблено три повноцінних додатка на основі технології блокчейн, на базі Етеріум. Проекти були створені в навчальних цілях, їхні сирцеві коди були виставлені у відкритий доступ, для того щоб допомогти іншим розробникам легше проектувати схожі додатки.

Література

[1] Ethereum Virtual Machine (EVM) Awesome List · ethereum/wiki Wiki [Електронний ресурс]: [Веб-сайт]. — Електронні дані. — Режим доступу: https://github.com/ethereum/wiki/wiki/Ethereum-Virtual-Machine-(EVM)-Awesome-List

[2] Solidity. Solidity 0.5.3 documentation [Електронний ресурс]: [Веб-сайт]. — Електронні дані. — Режим доступу: https://solidity.readthedocs.io/en/v0.5.3

[3] ERC20 standart [Електронний ресурс]: [Веб-сайт]. — Електронні дані. — Режим доступу: https://en.wikipedia.org/wiki/ERC-20

[4] web3.js library [Електронний ресурс]: [Веб-сайт]. — Електронні дані. — Режим доступу: https://web3js.readthedocs.io/en/1.0/

[5] Vue library [Електронний ресурс]: [Веб-сайт]. — Електронні дані. — Режим доступу: https://vuejs.org/

СТВОРЕННЯ ЧАТБОТІВ, ПОБУДОВА ДИЗАЙНУ ТА ВЗАЄМОДІЇ ЧАТБОТІВ З КОРИСТУВАЧАМИ. ДОСЛІДЖЕННЯ СИСТЕМ РОЗПІЗНАВАННЯ ТА ОБРОБКИ ТЕКСТУ

Ахмедов Абульфаз, Христина Спасник
Україна, Львів
Львівський національний університет імені Івана Франка
Факультет прикладної математики та інформатики
abulfazahmadov@gmail.com , hristinaspasnik@gmail.com

Вступ

У наш час все більше і більше набирають популярність соціальні мережі та месенджери, які забезпечують комунікацію між людьми. Разом із цим, також стають популярними так звані “чатботи” - програми, що на основі моделі машинного навчання, розпізнавання мови та нейронних мереж імітують розмову. Перші чатботи були створені ще в 60-х роках XX ст. , але з розвитком інтернет технологій та месенджерів, ця концепція почала стрімко набирати популярність у наш час. У цій роботі розглянуто спосіб побудови таких систем, створення UI/UX, дослідження сервісів, які дозволяються на основі http запитів, класифікувати текст користувача. З високого рівня робота бота полягає в тому, щоб мати можливість визначати найкращу відповідь для будь-якого отриманого повідомлення. Ця "найкраща"відповідь повинна (1) відповісти на запитання відправника, (2) надати відповідну інформацію відправнику, (3) поставити запитання щодо подальших питань, (4) продовжити розмову реалістичним способом. Це досить високий порядок. Бот повинен мати можливість зрозуміти наміри повідомлення відправника, визначити, який потрібно вибрати тип відповідного повідомлення (наступний запит, прямий відгук тощо), а також забезпечувати виконання правильних граматичних та лексичних правил під час формування відповіді.

Головна ідея

Ідея створення систем чатботів для месенджерів(на прикладі facebook messenger) базується на http запитах. У ролі клієнта виступає вебсайт або мобільна аплікація месенджера. Технічно все відбувається за схемою поданою на рисунку (Рис.1). У нас є клієнт — один із популярних месенджерів, що взаємодіє з користувачем, отримавши повідомлення від нього, формує та відправляє запит на сервер. Він отри-

мавши цей запит, обробляє його. Текст користувача, зазвичай, сервер відправляє на NLP сервіс, який, обробивши цей текст, повертає дані про нього назад. Тоді сервер, маючи усі ці дані про текст користувача, аналізує його, та формує відповідь, яку відправляє назад на месенджер, і там він передається користувачу. Створити чатбот, можна і без використання NLP сервісів, використовуючи тривіальну обробку повідомлення користувача на сервері. Застосування таких сервісів надає значну перевагу та більші можливості. За допомогою таких сервісів можна створити значно кращу систему, яка зможе відповідати на ширший набір питань. Таких сервісів внаслідок розвитку цієї індустрії є чимало. Кожен із них пропонує щось, що може стати в пригоді в конкретному випадку. Одним із найпопулярніших для NLP в наш час є Dialogflow, який раніше називався Api.ai. Цей продукт був куплений компанією Google та працює на основі машинного навчання та нейронних мереж. У цій системі варто зазначити кілька важливих для роботи із нею термінів, наприклад: Агент — об'єкт, що містить у собі всю інформацію для використання, та є одиницею, яку чатботи використовують для розпізнавання тексту. Саме розпізнавання та тренування агента відбувається на основі данних. Тренування відбувається за моделями Intent, Entity, про які більш детально буде розказано на самій презентації. Також агент покращується завдяки взаємодії з ним користувачів. На основі даних, які він отримує в процесі обробки текстів користувача, він теж навчається та стає кращим.

У нашому випадку, клієнтом виступає Facebook messenger або Telegram, який отримавши повідомлення від користувача , формує та відправляє запит на сервер, який після цього, обробляє його. Текст користувача, зазвичай, сервер відправляє на NLP(Natural Language Processing) сервер, для того, щоб зрозуміти намір користувача, та, отримавши дані про наміри користувача, відповідним чином формує ту чи іншу відповідь для користувача та відправляє назад до клієнту. Сам сервер реалізований за допомогою технології Node.js, який завдяки своїй архітектурі, що базується на подіях(Event driven architecture), є одним із найкращих інструментів для подібних додатків з міркувань швидкодії.

Головна ідея UI/UX чатботів

У чатботах, дуже важливо побудувати правильний хід взаємодії з користувачем. На основі досліджень та власного досвіду, ми рбимо висновок, що чим більш вузькою є спеціалізація чатбота, тим більше він був ефективним та точним. Дуже важливим у створенні прикладного програмного забезпечення будь-якого призначення, не тільки чатботів,

є забезпечення простої та інтуїтивно зрозумілої взаємодії з кінцевими користувачами продуктів. Основі дослідів, та опитувань, стало відомо те, що чим більше підказок чатбот надає користувачеві, тим кращою буде взаємодія із ним.

Література

[1] *https://docs.microsoft.com/en-us/azure/bot-service/bot-service-design-first-interaction*

[2] *https://dialogflow.com/docs/getting-started/basics*

РОЗРОБЛЕННЯ ГРАФІЧНОГО МОДУЛЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ПРОГНОЗУВАННЯ ПОГОДИ

Олег Блистів, Україна
Національний лісотехнічний університет України
Навчально-науковий інститут деревообробних та комп'ютерних технологій і дизайну

Вступ. Життя людей в значній мірі залежить від погодних умов та вміння їх прогнозувати. На сьогоднішній день прогнозування погоди проводиться на науковій основі з використанням потужних обчислювальних серверів і великого об'єму метеорологічних даних, які ними обробляються. Пронозовані параметри погоди різної завчасності відображають у мережі Інтернет на різноманітних сайтах у вигляді тексту, таблиць, карт, графіків.

Важливим залишається завдання постійного моніторингу та представлення метеорологічних даних прогнозу у формі, зручній для сприйняття користувачами у Інтернеті. Тому задача автоматичного контролю та відображення цих даних є актуальною задачею як в науковому так і в практичному плані.

Мета роботи полягає в розробленні графічного модуля інформаційної системи прогнозування погоди.

Виклад основного матеріалу. Якість подання числових даних в інформаційних системах істотним чином залежить від достовірності та способу їх представлення. При розробленні програмного забезпечення графічного модуля інформаційної системи прогнозування погоди, дані отримуються в реальному часі шляхом синтаксичного аналізу веб-ресурсу Українського гідрометеорологічного центру (https://meteo.ua/ua). З цією метою застосовано бібліотеку DiDom мови програмування PHP. Для пошуку необхідних HTML-елементів використовується метод:

find (параметр1, параметр2),

де параметр1 – назва CSS-селектора;
параметр2 – тип CSS-селектора (за замовчуванням Querry::TYPE CSS).

Отримані дані зберігаються у форматі JSON з метою забезпечення компактності та швидкої їх обробки.

Для графічної візуалізації погодних даних використано JavaScript-бібліотеку Chartist.js. Дана бібліотека застосовує формат векторної графіки SVG та підтримується всіма сучасними браузерами.

Розроблене програмне забезпечення представляє веб-сайт, головна сторінка якого представлена на рисунку 1.

Для відображення прогнозу погоди в певному населеному пункті України необхідно ввести його назву у полі пошуку у верхній лівій частині сторінки.

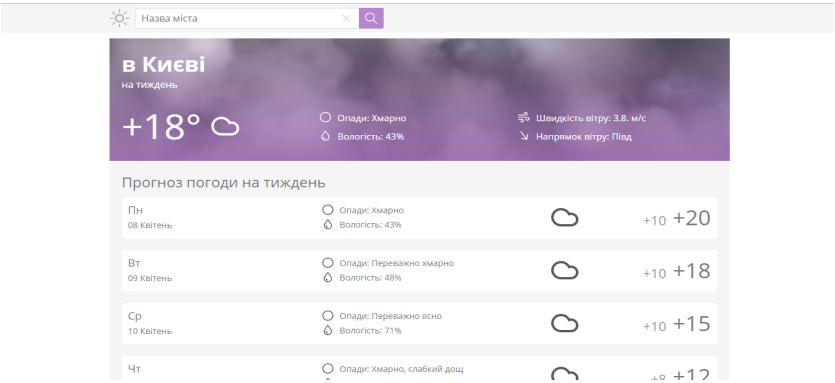


Рис. 1: Головна сторінка веб-сайту

Графічне представлення параметрів прогнозу (температури, вологості, тиску) на поточний тиждень демонструється в нижній частині цієї сторінки (рис. 2)

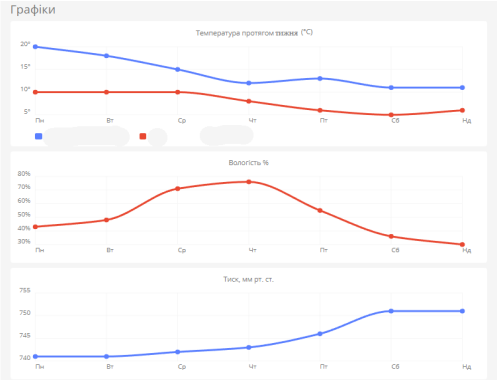


Рис. 2: Графічне представлення параметрів прогнозу на поточний тиждень

Для перегляду графіків параметрів прогнозу погоди на певний день (рис. 3) тижня потрібно вибрати його зі списку представленого на сторінці.

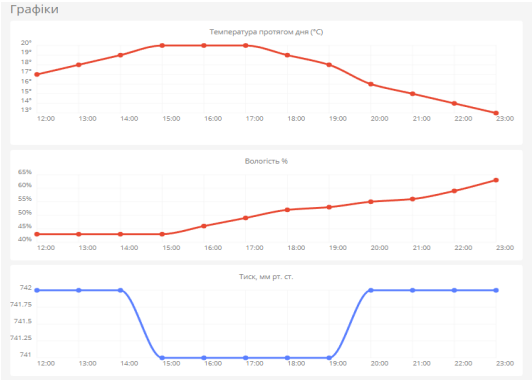


Рис. 3: Графіки параметрів прогнозу погоди на певний день тижня

Висновки. Розроблено програмне забезпечення графічного модуля інформаційної системи прогнозування погоди. Дану веб-систему можна застосовувати для отримання в зручному графічному вигляді детальної інформації про погодні умови у різних містах України.

Література

[1] DiDOM *[Електронний ресурс] : [Веб-сайт]. – Режим доступу: https://github.com/Imangazaliev/DiDOM (дата звернення 10.03.2019). – Назва з екрана*

[2] Chartist.js [Електронний ресурс] : [Веб-сайт]. – Режим доступу: https://gionkunz.github.io/chartist-js/ (дата звернення 17.03.2019). – Назва з екрана.

ЧИСЛОВЕ РОЗВ'ЯЗУВАННЯ ЗАДАЧІ АДВЕКЦІЇ-ДИФУЗІЇ ІЗ ЗАСТОСУВАННЯМ АЛГОРИТМУ ДІРІХЛЕ-НЕЙМАНА

Сподар Ірина Володимирівна, Флис Софія Тарасівна
Україна, Львів
Львівський національний університет ім. Івана Франка
факультет прикладної математики та інформатики
irynka-ww@ukr.net, sophia_sone4ko@ukr.net

Вступ

Числове дослідження процесів адвекції-дифузії відіграє важливу роль при математичному моделюванні задачі переносу забруднень у середовищі. Проте відомим є той факт, що застосування методу скінченних елементів (МСЕ) [1] до розв'язування цієї задачі у випадку великих чисел Пекле є неефективним з огляду на значні осциляції. В даній роботі застосовано алгоритм Діріхле-Неймана [2] з використанням спеціальних пробних функцій для усунення цієї проблеми. Для аналізу результатів здійснено верифікацію отриманого розв'язку.

Алгоритм Діріхле-Неймана

Розглянемо задачу адвекції-дифузії

$$\begin{cases} A_1 u_1 = div(bu_1) + b_0 u_1 = f \text{ в } \Omega_1, \\ A_2 u_2 = div(-\nu \nabla u_2 + bu_2) + b_0 u_2 = f \text{ в } \Omega_2, \\ u_1 = g_1 \text{ на } \partial\Omega_1 \backslash \Gamma, \\ u_2 = g_2 \text{ на } \partial\Omega_2 \backslash \Gamma, \\ u_1 = u_2 \text{ на } \Gamma, ` \\ -b \cdot n_\Gamma u_1 = \nu \frac{\partial u_2}{\partial n_\Gamma} - b \cdot n_\Gamma u_2 \text{ на } \Gamma. \end{cases}$$

Головна ідея методу Діріхле-Неймана полягає в тому, що на одній з підобластей формулюють задачу з крайовими умовами Діріхле, а на іншій – з крайовими умовами Неймана. Розв'язуючи ітераційно отримані задачі, знаходять послідовності функцій $\{{u_1}^{(k)}\} \in \Omega_1$, $\{{u_2}^{(k)}\} \in \Omega_2$ що збігаються до розв'язків вихідної гетерогенної задачі u_1 і u_2 відповідно. Співпадіння розв'язків u_1 і u_2 на спільній межі підобластей Ω_1 і Ω_2 забезпечується умовами спряження.

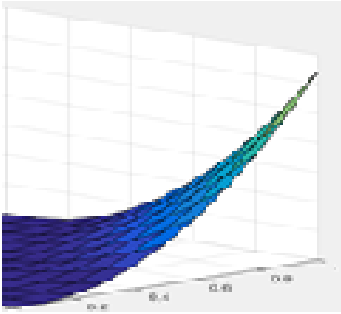


Рис. 1: Зображення пробних функцій $\psi_i(x_1, x_2)$

Вибір пробних функцій

Введемо спеціальні пробні функції [3]

$$\psi_i(x_1, x_2) = \phi_i(x_1, x_2) - \alpha\gamma(x_1, x_2),$$

$$\gamma(x_1, x_2) = w_n\phi_i(x_1, x_2)\phi_j(x_1, x_2)\phi_m(x_1, x_2),$$

де ϕ_i, ϕ_j, ϕ_m - білінійні базисні функції МСЕ, w_n - нормальна складова вектора швидкості, α - параметр, що дозволяє контролювати вплив допоміжної функції $\gamma(x_1, x_2)$. Введені модифіковані пробні функції зображені на рис. 1.

Верифікація результатів

Для верифікації розв'язку задаємо на двох протилежних границях області однорідну умову Неймана, а на двох інших – однорідну умову Діріхле. Розв'язок матиме вигляд циліндра, тобто на кожній з прямих $x_2 = a$, де $a \in [0, 2]$, він буде однаковим. Тому якщо тепер зафіксуємо координату x_2, то отримаємо аналог одновимірного випадку, який можемо порівняти з аналітичним розв'язком. З рис. 2, 3 бачимо, що розв'язки є дуже подібними.

Експериментальний порядок збіжності обчислимо за формулою:

$$p = \frac{ln(e_1) - ln(e_2)}{ln(h_1) - ln(h_2)}$$

Результати наведено в табл. 1.

Також здійснимо верифікацію розв'язку шляхом використання схеми Ейткена. Для експериментального порядку збіжності застосуємо формулу (1).

$$p = \frac{ln(\delta_1) - ln(\delta_2)}{ln(q)}. \tag{38}$$

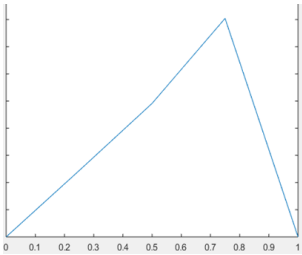


Рис. 2: Розв'язок на фіксованій прямій

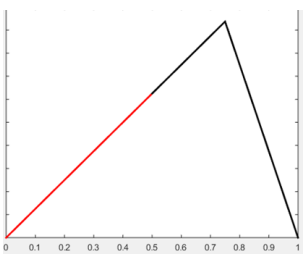


Рис. 3: Аналітичний розв'язок

	L_2	$W_2^{(1)}$
Апріорний порядок збіжності	2	1
Експериментальний порядок збіжності	2.02	0.93

Табл. 1: Порядок збіжності у порівнянні з аналітичним розв'язком

Результати наведені в табл. 2.

	L_2	$W_2^{(1)}$
Порядок збіжності	1.97	0.94

Табл. 2: Порядок збіжності за схемою Ейткена

Висновок

Розглянуто гетерогенну модель задачі адвекції-дифузії. Для знаходження розв'язку задачі застосовано алгоритм декомпозиції області Діріхле-Неймана з використанням модифікованих пробних функцій. Проведено верифікацію отриманого розв'язку шляхом порівняння його одновимірної проекції з аналітичним розв'язком та шляхом використання схеми Ейткена. Отримані результати демонструють, що експериментальний порядок збіжності близький до апріорного.

Література

[1] *Савула Я. Г.* Числовий аналіз задач математичної фізики варіаційними методами // Я. Г. Савула. - Львів: видавничий центр ЛНУ імені Івана Франка, Львів, 2004. – 221 с.

[2] *Quarteroni A.* Heterogeneous Mathematical Models in Fluid Dynamics and Associated Solution Algorithms // A. Quarteroni – NY : Springer, 2012. – p. 57. – (Multiscale and Adaptivity: Modeling, Numerics and Applications).

[3] *Fletcher C.* Computational Galerkin Methods // Springer-Verlag New York, 1984.– p. 352.

АДАПТИВНА СХЕМА МЕТОДУ ГРАНИЧНИХ ЕЛЕМЕНТІВ ДЛЯ ЗАДАЧА РОПОДІЛУ ЕЛЕКТРОСТАТИЧНОГО ПОТЕНЦІАЛУ

Фай Василь
Україна, Львів
Національний університет імені Івана Франка
Факультет прикладної математики та інформатики
vasylfay@gmail.com

У даній роботі представлено програмну реалізацію в середовищі MATLAB алгоритму для h-адаптивного методу граничних елементів(МГЕ) для розв'язування задач розподілу електростатичного потенціалу з використанням зваженої апостеріорної оцінки похибки кусково лінійних апроксимацій цього методу. Просторовий розподіл електричного потенціалу ϕ у напівпровіднику можна знайти шляхом розв'язку рівняння Пуассона

$$\Delta\varphi = -\frac{1}{\varepsilon\varepsilon_0}\rho(\vec{r}) \tag{39}$$

де ε – відносна діелектрична проникність матеріалу, ε_0 – електрична постійна, $\rho(r)$ густина об'ємного заряду, який визначається зарядом іонізованих домішок та вільними носіями. Для випадку невиродженої статистики вільних носіїв заряду

$$\rho(r) = -e\left[n_0 \exp\left(\frac{e\varphi}{kT}\right) - p_0 \exp\left(-\frac{e\varphi}{kT}\right) + N_A - N_D\right] \tag{40}$$

де n_0, p_0 – концентрації вільних електронів та дірок за умови відсутності поверхневого заряду, N_A, N_D – концентрації іонізованих акцепторів та донорів.

Рівняння (2) необхідно доповнити двома граничними умовами. Перша з них випливає із умови симетрії (посередині між порами потенціал повинен досягати екстремуму), а друга задає нормальну складову напруженості електричного поля на поверхні пори радіуса r^0

$$\left.\frac{d\varphi}{dr}\right|_{r=R} = 0 \tag{41}$$

$$\left.\frac{d\varphi}{dr}\right|_{r=r^0} = -\frac{\sigma}{\varepsilon\varepsilon_0} \tag{42}$$

Для дослідження задачі введемо загальний формолювання нашої задачі в наступному вигляді :

$$-\Delta u \underset{82}{=} f \text{ в } \Omega \tag{43}$$

$$\frac{\partial u}{\partial n} = \phi \text{ на } \Gamma \tag{44}$$

Розв'язання задачі (5)–(6) еквівалентне розв'язанню інтегрального рівняння вигляду:

$$u(x) = \tilde{N} f(x) + \tilde{V}\phi(x) - \tilde{K} g(x) \quad \text{for all } x \in \Omega \tag{45}$$

Спрямувавши x на границю отримаємо:

$$Wu = (1/2 - K')\,\phi - N_1 f \quad \text{on } \Gamma \tag{46}$$

V - потенціал простого шару, K - потенціал подвійного шару, K' спряжений оператор до K, W гіперсингулярний оператор, N_1 похідна по нормалі від оператора потенціалу Ньютона N_0

Інтегральне рівняння розв'язуємо методом Гальоркіна. Для апостеріорного оцінювання похибки МГЕ скористаємося відповідною зваженою похибкою запропоновану у роботі [1]:

$$\widetilde{\varrho}_\ell := \left\| h_\ell^{1/2} \left(WU_\ell - (1/2 - K')\,(\Phi_\ell + \Lambda_\ell) \right) \right\|_{L_2(\Gamma)} \tag{47}$$

Де Λ_ℓ є розв'язком такого допоміжного рівняння

$$\langle V\Lambda_\ell, \Psi_\ell \rangle_\Gamma = \langle N_0 F_\ell, \Psi_\ell \rangle_\Gamma \tag{48}$$

Для локального адаптування сітки скористаємося алгоритмом Дюрфнера. Обчислимо значення похибки за формулою (9) для кожного скінченного елемента, визначимо множину елементів якої виконується умова:

$$\theta \widetilde{\varrho}_\ell^2 = \theta \sum_{\tau \in \mathcal{E}_\ell \cup T_\ell} \widetilde{\varrho}_\ell(\tau)^2 \le \sum_{\tau \in \mathcal{M}_\ell} \widetilde{\varrho}_\ell(\tau)^2 \tag{49}$$

θ – параметр адаптації, $\mathcal{E}$–сітка методу, $\mathcal{M}$ - підмножина елементів для перерахунку. Після знаходження даної множини, подрібнимо кожен елемент для задання нової сітки.

Числові результати

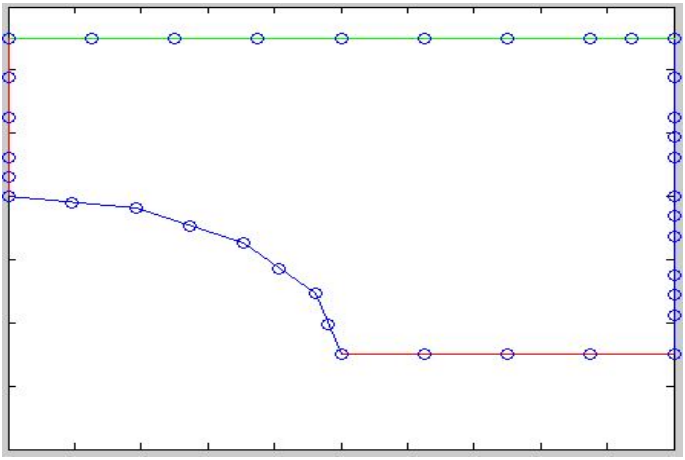
Для перевірки роботи програмного забезпечення розглянемо таку спрощенну тестову задачу :

$$-\Delta u = 1 \text{ в } \Omega \tag{50}$$

$$\frac{\partial u}{\partial n} = \begin{cases} 0, & \text{на } \Gamma_R \\ 5, & \text{на } \Gamma_{r^0} \end{cases} \tag{51}$$

На рисунку показана сітка отримана при сітці з 30 елементів.

Табл. 1 демонструє результати, одержані в певних точках всередині області при згущенні сітки за допомогою h-адаптивної схеми.



елементів	(0,0)	(0,0)	(0,0)	(0,0)	(0,0)
10	2.3770	2.1987	2.0045	1.7890	1.5584
50	2.3774	2.1990	2.0047	1.7892	1.5586
150	2.3774	2.1990	2.0047	1.7892	1.5585
500	2.3775	2.1991	2.0048	1.7893	1.5585

Табл. 1

Література

[1] *C. Carstensen, E.P. Stephan* A posteriori error estimates for boundary element methods, Math. Comp. 64 (1995), 483–500.

[2] *O.C. Zienkiewicz, J. Z. Zhu* A Communications in Applied Numerical Methods, 1988

[3] *Л.С. Монастирський, І.Б. Оленич, Б.С. Соколовський* Моделювання розподілу електростатичного потенціалу у поруватому кремнії, Львівський національний університет імені Івана Франка, 2016

МІКРОСЕРВІСНА АРХІТЕКТУРА ПРИ РОЗРОБЦІ ТА МОДЕРНІЗАЦІЇ ВЕБ-ЗАСТОСУНКІВ

Роман Мулик
Україна, Львів
Львівський національний університет імені Івана Франка
Факультет прикладної математики та інформатики
work.mulykr@gmail.com

Слідкуючи за трендами та тенденціями розвитку методів розробки програмного забезпечення, можна зауважити, що саме мікросервісна архітектура стає дуже популярною, оскільки надає оптимальне вирішення для побудови веб-аплікацій, які легко підтримувати та розширювати.

Будуючи веб-аплікацію з використанням принципів мікросервісної архітектури (MSA), розробник розширює простір своїх можливостей, адже кожен компонент (мікросервіс) такого додатку є незалежними, крім того усі модулі можуть бути написаними на різних мовах програмування.

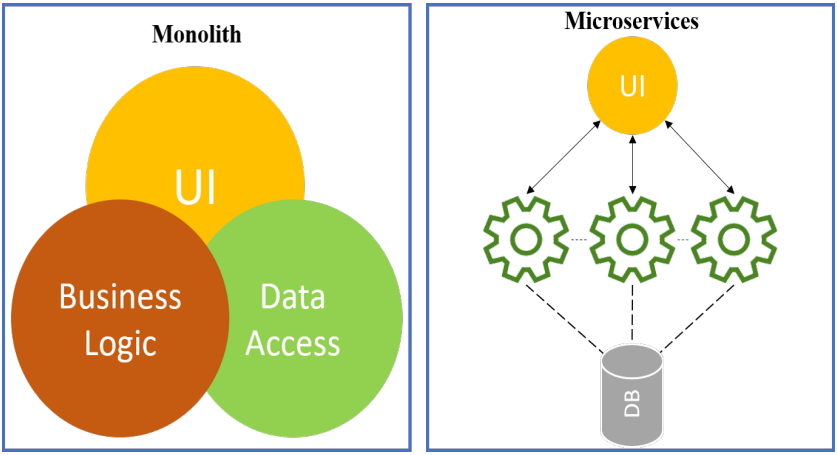


Рис. 1: Архітектури

Головним завданням мікросервісу є надати клієнту певний API (у випадку веб-аплікації WebAPI), який буде обробляти запити користувача та надавати відповіді в очікуваному форматі.

Важливим є те, що мікросервіс має виконувати лише конкретну “одиницю” роботи і виконувати її досить якісно. Самі ж сервіси теж можуть взаємодіяти за допомогою простих систем обміну повідомленнями.

Побудову аплікації (в різних варіаціях) з використанням стратегії MSA продемонстровано на прикладі створення інтернет-магазину. Протягом розробки вищезгаданого додатку, пройдено міграцію від MVC ASP.NET Core 2.1 аплікації до чистої MSA з використанням технології SPA (Angular 7), проаналізовано шляхи вирішення проблем, які виникають у ході такої міграції. Отримуючи гібридні архітектури, проаналізував переваги, недоліки та доцільність таких маніпуляцій.

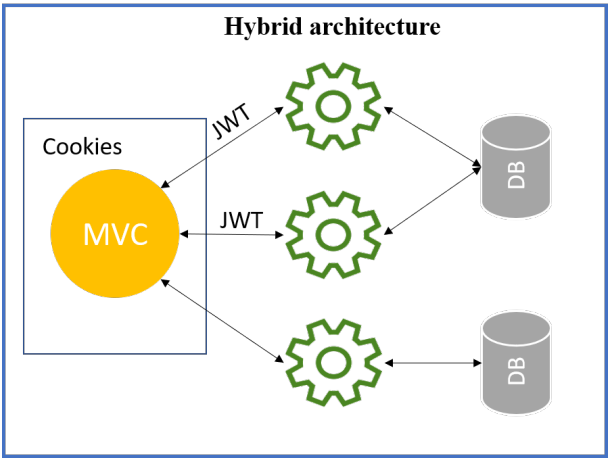


Рис. 2: Гібридна архітектура

Така тема є доволі актуальною, оскільки багато аплікацій доступних у мережі Інтернет є розробленими саме за шаблоном MVC, тому мати відомості щодо осучаснення їх шляхом переведення на модерні та більш “легкі” архітектури є щонайменше корисним, а взагалі важливим.

Адаптація MVC аплікації під MSA дає й широкий спектр переваг при її публікації в Інтернеті. Серед них такі як висока доступність, масштабованість окремих компонент аплікації, можливість повторного використання сервісів, проста заміна модулів, та менша вартість утримування її на просторах Інтернету. Серед недоліків – ряд проблем пов’язаних з визначенням прав доступу до ресурсів у певних випадках.

Література

[1] *Adam Freeman* Pro ASP.NET Core MVC 2.

[2] https://microservices.io/