

Doctoral Courses 2023-2024 I. (fall) semester

Courses in black: only for Hungarian students

Courses in green: for all students

- 1. Kertész Attila: IoT-Felhő rendszerek szimulációja / Simulating IoT-Cloud systems**
- 2. Bilicki Vilmos, Siket István: Prompt Engineering for Software Development**
- 3. Horváth Péter: Mikroszkópiai képelemzés / Microscopical image analysis**
- 4. Balogh János, Békési József: Pakolási algoritmusok / Packing type problems and algorithms**
- 5. Kincses Zoltán, Schäffer László: FPGA alapú beágyazott rendszerek tervezése / FPGA-based embedded system design**
- 6. Vágvölgyi Sándor: Termátíró rendszerek / Term rewriting systems**
- 7. Nyúl László: Fuzzy módszerek a képfeldolgozásban / Use of Fuzzy Methods in image Processing**

Kertész Attila: IoT-Felhő rendszerek szimulációja / Simulating IoT-Cloud systems

Leírás: Napjainkban a számítási felhők egyre nagyobb teret hódítanak az internetes szolgáltatások körében. A legelterjedtebb felhő szolgáltatók felismerték az új technológiák, így pl. az IoT (Internet of Things) támogatás szükségességét, és rendelkeznek is bizonyos szintű IoT alkalmazásfejlesztő megoldásokkal. A köd paradigma kiegészíti a felhő erőforrásokat a felhasználóhoz közel elhelyezett, korlátozott kapacitású csomópontokkal, amelyeken nagy mennyiségű adatok hatékony feldolgozása és elemzése könnyebben megvalósítható, valós idejű IoT alkalmazások esetén is. Hatékony IoT-Köd-Felhő rendszerek tervezésének és üzemeltetésének komoly anyagi vonzata lehet, hiszen számtalan, a rendszert leíró paraméter (VM tulajdonságok, hálózati jellemzők) és erőforrásmenedzselő algoritmus (ütemező, tehermentesítő) finomhangolására van szükség. Ezért a kutatók körében elterjedtek a különböző szimulációs megoldások, amelyeket képesek realiztikusan modellezni és költséghatékonyan vizsgálni ezeket a rendszereket. Ilyen szimulációs megoldás a DISSECT-CF-Fog is, amely részletes modellel rendelkezik IoT rendszerek szimulálásra: képes szenzorokat, aktuátorokat és IoT eszközöket modellezni, illetve tetszőleges IoT alkalmazások viselkedését elemezni, amely köd és felhő erőforrásokat használ.

Tematika:

A bemutatásra kerülő főbb témakörök:

- Számítási felhők kialakulása, fajtái, tulajdonságai
- IoT rendszerek kialakulása, elemei, tulajdonságai
- A Cloud-to-Thing rendszerek jellemzői, felépítése
- Szimulációs megoldások
- A DISSECT-CF és DISSECT-CF-Fog szimulátor
- Ütemezési és tehermentesítő stratégiák
- Mobilitási és migrációs stratégiák

Description: Nowadays, Cloud Computing has become one of the most trending services on the Internet. Cloud service providers recognised the need to support new technologies such as the Internet of Things (IoT) and various solutions for IoT application development. However, when using only clouds, the overloaded communication channels and increased response time can degrade the performance of IoT services. Fog Computing manages nodes with limited resource capacity, located close to end-users, providing efficient processing and analysis of large amounts of real-time IoT applications. The design and operation of efficient IoT-Fog-Cloud systems can have significant financial implications, as they can only be operated economically by fine-tuning a large number of parameters (VM properties, network characteristics) and resource management algorithms (scheduler, offloading). Therefore, various simulation solutions are widely used by researchers to realistically model and cost-effectively test these systems. One such simulation solution is DISSECT-CF-Fog, which provides a detailed model for simulating IoT systems: it can model sensors,

actuators and IoT devices, and analyse the behaviour of arbitrary IoT applications using fog and cloud resources.

The material of the course covers the following topics:

- Characterizing Cloud Computing: history, origins, types, properties
- IoT systems: origins, elements, properties
- Characteristics and structure of the Cloud-to-Thing Continuum
- Various simulation approaches
- The DISSECT-CF and the DISSECT-CF-Fog simulator
- Scheduling and offloading strategies
- Mobility and migration strategies

Literature:

- A. Kertész, Characterizing cloud federation approaches. In: Cloud computing: challenges, limitations and R&D solutions. Computer communications and networks. Springer, Cham, pp. 277-296, 2014.
- J. Gubbi, R. Buyya, S. Marusic, M. Palaniswami. Internet of Things (IoT): A vision, architectural elements, and future directions. Future Generation Computer Systems, Volume 29, Issue 7, pp. 1645--1660, September 2013.
- A. Markus and A. Kertesz. A Survey and Taxonomy of Simulation Environments Modelling Fog Computing. Simulation Modelling Practice and Theory, Volume 101, 2020.

Bilicki Vilmos, Siket István: Prompt Engineering for Software Development

Description: This course explores the principles and techniques of prompt engineering for software development, focusing on utilizing large language models like GPT-4 and other open source models. Students will learn how to effectively design prompts for various software development tasks, such as code generation, bug identification, documentation, and code review. The course also covers topics like fine-tuning and transfer learning, AI infrastructure.

Topics:

1. Introduction to Language Models in Software Development: An overview of how large-scale language models like GPT-4 are utilized in the software development process, covering their capabilities, limitations, and underlying architectures.
2. Basics of Prompt Engineering for Developers: The importance of crafting effective prompts for generating useful, accurate, and relevant responses from language models in software development applications.
3. Prompt Types and Techniques for Developers: Exploration of various types of prompts and techniques for crafting effective prompts, with a focus on applications in software development, such as code generation and bug identification.
4. Open Source Large Language Models in Software Development: A survey of popular open source large language models, such as BERT, RoBERTa, and T5, and their applications in software development tasks.
5. Model Fine-tuning and Transfer Learning for Software Tasks: Techniques for adapting pre-trained language models to specific software development tasks using transfer learning and fine-tuning, including code generation, documentation, and code reviews.
6. Evaluating Model Performance in Software Development: Methods and metrics for assessing the quality and effectiveness of language model outputs in software development, including both quantitative and qualitative analyses.
7. Bias and Fairness in Language Models for Software Development: Understanding and addressing biases present in large language models and their implications for software development, with a focus on ensuring fairness and inclusivity in generated code and recommendations.
8. Infrastructure for AI-powered Software Development: An overview of the infrastructure required for AI-driven software development, including vector stores, distributed computing, and storage solutions for efficient language model utilization.
9. Code Review and Collaboration with AI: Techniques for leveraging language models in collaborative code review processes, including detecting issues, suggesting improvements, and enhancing overall code quality.
10. Advanced Prompt Engineering Techniques for Developers: In-depth exploration of techniques for optimizing prompts in software development contexts, such as enhancing code generation, improving error detection, and facilitating code refactoring.

11. Domain-specific Prompt Engineering in Software Development: Strategies and examples for crafting prompts specific to various software development domains, such as web development, mobile development, data science, and game development.
12. Interactive Applications and Conversational AI for Developers: Designing prompts for interactive software development applications, such as AI-assisted code editors, developer chatbots, and conversational agents for user support.
13. Data Collection and Dataset Curation for Software Development: Techniques for creating, collecting, and curating datasets to train language models and improve their performance on software development tasks.
14. Final Project Presentations and Course Wrap-up: Students present their final projects, showcasing their prompt engineering skills applied to real-world software development problems, followed by a course review and discussion on future developments in the field.

Horváth Péter: Mikroszkópai képelemzés / Microscopical image analysis

Tömbösített kurzus, 4 alkalom

Tematika:

High content mikroszkópia

Képfeldolgozás alapjai

Objektum felismerés, szegmentálás, 2D és 3D

Klasszikus módszerek, mélytanulók, jellemzők kinyerése

Szakirodalom:

<https://www.nature.com/articles/s41587-022-01302-5>

<https://arxiv.org/pdf/1505.04597.pdf>

<https://www.nature.com/articles/nrd.2018.99>

<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8247631/>

<https://arxiv.org/pdf/1703.06870.pdf>

Balogh János, Békési József: Pakolási algoritmusok / Packing type problems and algorithms

Leírás:

A ládapakolási feladat változatai (és a rokon pakolási feladatok) általában NP-nehéz kombinatorikus optimalizálási problémák. Emiatt ezek közelítő (approximációs) algoritmusainak vizsgálata került előtérbe a kutatásokban. A legrosszabb eset vizsgálatok esetén az ezt vizsgáló algoritmusok teljesítményének mérése offline esetben az approximációs hányados, online esetben a versenyképességi hányados segítségével történhet. Ezen teljesítménymutatók aszimptotikus változatai elterjedtek, amelyek jobban bevett módszerek, jobban utalnak az algoritmus teljesítményére a legrosszabb esetben.

Megismerünk néhány ládapakolási problémát és megoldási módszert, a feladat különböző változatait az erre adaptált algoritmusokkal, illetve néhány rokon és kitekintésként tárgyalt pakolási problémát és algoritmust. A hangsúly a ládapakolási feladatokon és az online változataikon lesz, de ennek offline változata is szóba kerül, valamint egyes rokon feladatok is: klasszikus approximációs és tipikus online feladatok, hátizsákfeladat, stb.

Tematika:

Szabási feladatok. Lineáris (egészértékű) programozási megfogalmazások.

Approximációs algoritmusok.

Online algoritmusok (tipikus online feladatok).

Ládapakolási feladatok.

Offline és online feladat.

A legrosszabb-eset viselkedés, és az algoritmusok teljesítménymutatói.

Klasszikus online ládapakolási algoritmusok és alsó korlát eredmények. First Fit, Next Fit, Best Fit, és általában az Any Fit család.

Félig online ládapakolási feladatok. First Fit decreasing (FFD).

A ládapakolás változatai és egyéb egzotikus feladatai; elemszámkorlátos pakolás, vektorpakolás, többdimenziós pakolás, változó méretű ládapakolás, ládapakolás túltöltéssel, ládafedés, fekete-fehér, köteget és klaszteres ládapakolás, dinamikus ládapakolás, stb. (LIB-pakolás, szcenárió alapú pakolások).

Kapcsolódó hátizsák-problémák.

Kapcsolódó ütemezési problémák; ládanyújtás (bin stretching).

Kitekintés: egyéb pakolási feladatok: Egymást követő négyzetek pakolása, háromszög-pakolás, körpakolás, számsorozatok kupacokba pakolása.

Kötelező és ajánlott irodalom:

- Bernhard Korte and Jens Vygen, Combinatorial Optimization Theory and Algorithms, ISSN 0937-5511, ISBN 978-3-662-21710-8, Springer-Verlag Berlin Heidelberg New York, 2000.
- E.G. Coffman, Jr., M.R. Garey, D.S. Johnson, Approximation algorithms for bin packing: A survey, Approximation Algorithms for NP-hard problems, Boston, 1996, Chapter 2, pp. 46-93.

**Kincses Zoltán, Schäffer László: FPGA alapú beágyazott rendszerek tervezése /
FPGA-based embedded system design**

Tömbösített kurzus

Előfeltétel: FPGA-alapú rendszerek tervezése MSc kurzus

Tematika:

FPGA SoC áramkörök felépítése

FPGA alapú beágyazott rendszerek tervezése, megvalósítása, tesztelése

Xilinx Vivado, Xilinx Vitis

FPGA alapú magas szintű szintézis

Xilinx Vitis HLS

Szakirodalom:

Xilinx Vivado User Guide

Xilinx Vitis High-Level Synthesis User Guide

Vágvölgyi Sándor: Termátíró rendszerek / Term rewriting systems

A PhD hallgatók a mester szakos hallgatók feladatain felül egy-egy előadást tartanak a tananyagból.

Tematika

Absztrakt redukáló rendszerek. Termátíró rendszerek. Term egyenlőség rendszerek.

Megálló (termináló) tulajdonság. A probléma eldönthetlensége. A megállás bizonyítása speciális esetekben rendezések segítségével. Az interpretációs módszer. Egyszerűsítő rendezések.

Összefolyó (konfluens) tulajdonság. A probléma eldönthetlensége. Kritikus pár. Ortogonális termátíró rendszerek.

Lezárási algoritmus. Az alapvető lezárási algoritmus. A fejlesztett lezárási algoritmus. Huet lezárási algoritmusa.

Ground termátíró rendszerek. Redukált ground termátíró rendszerek. Kapcsolat faautomatákkal. Ground term egyenlőség rendszerek.

Kiterjesztések. Átírás modulo ekvacionális elméletek. Rendezett átírás. Feltételes átírás. Magasabb szintű átíró rendszerek. Redukáló stratégiák. Szűkítés.

Ajánlott irodalom:

F. BAADER, T. NIPKOW, Term Rewriting and All that, Cambridge University Press, 1998.

E. OHLEBUSCH, Advanced Topics in Term Rewriting, Springer Verlag, 2002.

Nyúl László: Fuzzy módszerek a képfeldolgozásban / Use of Fuzzy Methods in image Processing

Tematika:

Fuzzy halmazok, műveletek, fuzzy logika
Fuzzy halmazok tulajdonságai
Fuzzy képfeldolgozó rendszerek felépítése
Fuzzy képjavítási módszerek
Fuzzy éldetektálás és elősszekötés
Fuzzy képszegmentálás (klaszterezés, kNN, c-means)
Fuzzy összefüggőség és változatai, algoritmusai
Fuzzy összefüggőség alkalmazása orvosi képek szegmentálásában

Topics:

Imperfection is inherently present in most image processing and image analysis problems. It may be in the image data, e.g. due to the acquisition device and process, noise, discretization artifacts, and inhomogeneity of the subject of imaging. On the other hand, in many real life image understanding applications the objectives and the expert knowledge can only be expressed in vague terms. Fuzzy set theory allows formally handling vague terms, and reasoning with degrees of truthfulness and falsehoods. Fuzzy logic is successfully applied in many fields, from control theory to pattern recognition and artificial intelligence. Expert systems, such as those in medical diagnostics also can benefit from fuzzy set theory. Fuzzy image processing is the collection of approaches that represent and process images, their segments and features as fuzzy sets. In this course we cover the basics of fuzzy set theory and fuzzy logic, and discuss, through examples, how fuzzy set representation can be applied in image processing at low-level (pixels), mid-level (image segments), and high-level (objects and scenes) tasks.

Literature:

- James C. Bezdek, James Keller, Rangu Krishnapuram, Nikhil R. Pal: Fuzzy Models and Algorithms for Pattern Recognition and Image Processing, Kluwer Academic Publishers, 1999.
- Scientific journal papers