Planning and Optimization
A1. Organizational Matters

Malte Helmert and Thomas Keller

Universität Basel

September 18, 2019

A1.1 People & Coordinates

A1.2 Target Audience & Rules

A1.3 Course Content

A1.1 People & Coordinates

People: Lecturers

Malte Helmert   Thomas Keller

Lecturers

Malte Helmert
- ▶ email: `malte.helmert@unibas.ch`
- ▶ office: room 06.004, Spiegelgasse 1

Thomas Keller
- ▶ email: `tho.keller@unibas.ch`
- ▶ office: room 04.005, Spiegelgasse 1

People: Assistants

Salomé Eriksson   Florian Pommerening   Silvan Sievers

Assistants

Salomé Eriksson
- ▶ email: `salome.eriksson@unibas.ch`
- ▶ office: room 04.001, Spiegelgasse 5

Florian Pommerening
- ▶ email: `florian.pommerening@unibas.ch`
- ▶ office: room 04.005, Spiegelgasse 1

Silvan Sievers
- ▶ email: `silvan.sievers@unibas.ch`
- ▶ office: room 04.002, Spiegelgasse 1

Time & Place

Lectures
- ▶ time: Mon 14:15-16:00, Wed 14:15-16:00
- ▶ place: room 00.003, Spiegelgasse 1

Exercise Sessions
- ▶ time: Wed 16:15-18:00
- ▶ place: room 00.003, Spiegelgasse 1

first exercise session: today

Planning and Optimization Course on the Web

Course Homepage
`https://dmi.unibas.ch/en/academics/computer-science/`
`courses-in-fall-semester-2019/`
`lecture-planning-and-optimization/`
- ▶ course information
- ▶ slides
- ▶ exercise sheets and materials
- ▶ bonus materials (not relevant for the exam)

registration:
- ▶ `https://services.unibas.ch/`
- ▶ Please register today to receive all course-related emails!

A1.2 Target Audience & Rules

Target Audience

target audience:
- ▶ M.Sc. Computer Science
  - ▶ Major in Machine Intelligence:
    module Concepts of Machine Intelligence
  - ▶ Major in Distributed Systems:
    module Applications of Distributed Systems
- ▶ M.A. Computer Science ("Master-Studienfach")
  module Concepts of Machine Intelligence
- ▶ other students welcome

Prerequisites

prerequisites:
- ▶ general computer science background: good knowledge of
  - ▶ algorithms and data structures
  - ▶ complexity theory
  - ▶ mathematical logic
  - ▶ programming
- ▶ background in Artificial Intelligence:
  - ▶ Foundations of Artificial Intelligence course (13548)
  - ▶ in particular chapters on state-space search

Gaps?
⤳ talk to us to discuss a self-study plan to catch up

Exam

- ▶ written examination (120 min)
- ▶ date: January 27
- ▶ 8 ECTS credits
- ▶ admission to exam: 50% of the exercise marks
- ▶ final grade based on exam exclusively
- ▶ no repeat exam

Exercise Sheets

exercise sheets (homework assignments):
- ▶ solved in groups of at most three ($3 < 4$),
  submitted via Courses
- ▶ project-oriented assignments
  - ▶ each exercise sheet covers one part of the lecture
    (on average 2 weeks = 8 chapters)
  - ▶ substantial in scope ⤳ don't start too late
  - ▶ handed out at beginning of each part
  - ▶ work on these while we cover this part in the lecture
  - ▶ due Sunday after the end of the part
  - ▶ scope and marks proportional to covered topics
- ▶ mixture of theory, programming and experiments
- ▶ research aspects ⤳ be independent, but ask questions!

Programming Exercises

programming exercises:

- ▶ part of regular assignments
- ▶ solutions that obviously do not work: 0 marks
- ▶ work with existing C++ and Python code
- ▶ Vagrant virtual machine running Linux
  - ▶ can do your own native setup (but no support!)
- ▶ pull from Mercurial (hg) repository

Exercise Sessions

exercise sessions:

- ▶ discuss past homework assignments
- ▶ ask questions about current assignments (and course)
- ▶ work on homework assignments
- ▶ sometimes live exercises

Plagiarism

Plagiarism (Wikipedia)

Plagiarism is the "wrongful appropriation" and "stealing and publication" of another author's "language, thoughts, ideas, or expressions" and the representation of them as one's own original work.

consequences:

- ▶ 0 marks for the exercise sheet (first time)
- ▶ exclusion from exam (second time)

if in doubt: check with us what is (and isn't) OK before submitting

exercises too difficult? we are happy to help!

A1.3 Course Content

Learning Objectives

Learning Objectives
- ▶ get to know theoretical and algorithmic foundations
  of classical & probabilistic planning
  and work on practical implementations
- ▶ understand fundamental concepts underlying modern planning
  algorithms and theoretical relationships that connect them
- ▶ become equipped to understand research papers
  and conduct projects in this area

Course Material

course material:
- ▶ slides (online + printed handouts)
- ▶ no textbook
- ▶ additional material on request

Hands-On Sessions

- ▶ This week and next week, we are running hands-on sessions
  in the exercise time slot (Wed 16:15-18:00).
- ▶ We hope you received our emails about
  - ▶ bringing your laptop
  - ▶ setting up VirtualBox and Vagrant
- ▶ If not, no problem: just team up with others.
- ▶ But please set up the software environment as soon
  as possible: you will need it for the exercises.

Priming the Virtual Machine

Assumptions: VirtualBox and Vagrant installed
VirtualBox: `https://www.virtualbox.org`

Vagrant: `https://www.vagrantup.com`

on Ubuntu 18.04: `sudo apt install virtualbox-qt vagrant`

One-time setup of the Virtual Machine
Download the `Vagrantfile` from the course homepage
and put it into an empty directory.

Open a console in that directory and execute `vagrant up`.
(This can take quite a long time.)

Logging in to the Virtual Machine
Open a console in the directory with the Vagrantfile
and execute `vagrant ssh`.

Demo

- ▶ We also use the virtual machine for demos during the lecture.
- ▶ To reproduce these, log into the virtual machine and go to the base directory for the course:

Base Directory for Demos
```
$ cd /vagrant/planopt-hs19
```

Under Construction. . .

- ▶ Advanced courses are close to the frontiers of research and therefore constantly change.
- ▶ We are always happy about feedback, corrections and suggestions!