Master program in Computational Physics, Mathematics and Life Science

Tom Andersen[1]

Andreas Austeng[2]

Arne Bang Huseby[3]

Michele Cascella[4]

Marianne Fyhn[1]

Morten Hjorth-Jensen[5]

Hans Petter Langtangen[2,6]

Katrine Langvad (admin)[5]

Anders Malthe-Sørenssen[5]

Kend-Andre Mardal[3]

Knut Mørken[3]

Grete Stavik-Døvle (admin)[5]

Joakim Sundnes[2,6]

Marte Julie Sætra (student representative)[5]

[1]Department of Biosciences, University of Oslo
[2]Department of Informatics, University of Oslo
[3]Department of Mathematics, University of Oslo
[4]Department of Chemistry, University of Oslo
[5]Department of Physics, University of Oslo
[6]Simula Research Laboratory

October 2015

Overarching description of the CPMLS program

Students of this program learn to use the computer as a laboratory for solving problems in science and engineering. The program offers exciting thesis projects from many disciplines: biology and life science, chemistry, mathematics, informatics, physics, geophysics, mechanics, geology, computational finance, computational informatics, bid data analysis, digital signal processing and image analysis – the candidates select research field according to their interests.

A Master's degree from this program gives the candidate a methodical training in planning, conducting, and reporting large research projects, often together with other students and university teachers. THe projects emphasize finding practical solutions, developing an intuitive understanding of the science and the scientific methods needed to solve complicated problems, use of many tools, and not least developing own creativity and independent thinking. The thesis work is a scientific project where the candidates learn to tackle a scientific problem in a professional manner. The program aims also at developing a deep understanding of the role of computing in solving modern scientific problems. A candidate from this program gains deep insights in the fundamnetal role computations play in our advancement of science and technology, as well as the role computations play in society.

Description of learning outcomes

The power of the scientific method lies in identifying a given problem as a special case of an abstract class of problems, identifying general solution methods for this class of problems, and applying a general method to the specific problem (applying means, in the case of computing, calculations by pen and paper, symbolic computing, or numerical computing by ready-made and/or self-written software). This generic view on problems and methods is particularly important for understanding how to apply available, generic software to solve a particular problem.

Computing competence represents a central element in scientific problem solving, from basic education and research to essentially almost all advanced problems in modern societies. Computing competence is simply central to further progress. It enlarges the body of tools available to students and scientists beyond classical tools and allows for a more generic handling of problems. Focusing on algorithmic aspects results in deeper insights about scientific problems.

A candidate with a Master of Science degree from this program

- has deep knowledge of the scientific method at an advanced level
    1. This implies the ability to understand advanced scientific results in new fields
    2. To gain a fundamental understanding of the methods and tools which are necessary
    3. Can develop and apply advanced computational methods to scientific problems

4. Is capable of judging and analyzing all parts of the obtained scientific results
5. Is able to improve upon exisiting algorithms and develop proper error estimates and critically assess the validity of the results
6. To present these results orally and in written form as scientific reports/articles.
7. Critically evaluate and analyze scientific data and error sources
8. Can propose new hypotheses and suggest solution paths
9. Can generalize mathematical algorithms and apply them to new situations
10. For more applied directions, to be able to link the numerical results with specific applications and/or experimental data
11. To be able to develop mathematical models and algorithms in order to describe experimental data
12. To understand at a deeper level how to make science reproducible and how to link this to a sound ethical scienfitic conduct

- has a deep understanding of what computing means, entailing several or all of the topics listed here

    1. To develop a deep understanding of how computing is used to solve scientific problems
    2. Deep knowledge of the most fundamental algorithms involved, how to optimize these and understand statistical uncertainty quantifications.
        (a) Overview of advanced algorithms and how they can be accessed in available software and how they are used to solve scientific problems.
    3. Knowledge of high-performance computing elements: memory usage, vectorization and parallel algorithms.
    4. Understanding of approximation errors and what can go wrong with algorithms
    5. Knowledge of at least one computer algebra system and how it is applied to perform classical mathematics (calculus, linear algebra, differential equations - with verification).
    6. Extensive experience with programming in a high-level language (MATLAB, Python, R). Experience with programming in a compiled language (Fortran, C, C++).
    7. Experience with implementing and applying numerical algorithms in reusable software that acknowledges the generic nature of the mathematical algorithms.

8. Experience with debugging software, e.g., as part of implementing comprehensive tests.
9. Experience with programming of testing procedures.
10. Experience with different visualization techniques for different types of computed data.
11. Critical evaluation of results and errors using various testing environments
12. Be able to develop software and algorithms for solving complicated scientific problems independently and/or in collaboration with other students.
    (a) To be able to design a maintainable program in a systematic way, using tools such as version control and modern scripting languages, enhances thereby the understanding of software carpentry.
13. To understand how to increase the efficiency of numerical algorithms and pertinent software.
14. To be able to use effeciently high-performance computing resources, from compilers to hardware architecture.
15. To be able to develop stringent requirements to efficiency and precision of software.
16. To understand and use proper tools to make science reproducible, as well as developing a sound ethical approach to scientific problems.

- Develops professional competence through the thesis work, entailing
  1. Mature professionally and be able to work independently
  2. Can communicate in a professional way scientific results, orally and in written form
  3. Can plan and complete a research project.
  4. Develop one's algorithmic thinking for gaining deeper insights about scientific problems.

- Develope virtues, values and attitudes that lead to a better understanding of ethical aspects of the scientific method, as well as promoting central aspects of the scientific method to society. This means for example
  1. The candidate can reflect on and develop strategies for making science reproducible and to promote the need for a proper ethical conduct.
  2. Has a deep understanding of the role basic and applied research and computing play for progress in society.
  3. To be able to promote, use and develop version control tools in order to make science reproducible.

4. To be able to critically evaluate the consequences of own research and how this impacts society.
5. To mature an understanding of the links between basic and applied research and how these shape, in a fundamental way, progress in science and technology.
6. To develop an understading of the role research and science can play together with industry and society in general.

By completing a Master of Science thesis you will have developed a critical understanding of the scientific methods you have studied, a better understanding of the scientific process per se as well as having developed perspectives for future work and how to verify and validate your work.

Admission critera

The following higher education entrance qualifications are needed

- A completed bachelor's degree (undergraduate) comparable to a Norwegian bachelor's degree in one of the following disciplines
  1. Biology, molecular biology, biochemistry or any life science degree
  2. Physics, astrophysics, astronomy, geophysics and meteorology
  3. Mathematics, mechanics, statistics and computational mathematics
  4. Computer science and electronics
  5. Chemistry
  6. Materials Science and nanotechnology
  7. Any undergraduate degree in engineering
  8. Mathematical finance and economy
  9. Economy
- For international students, an internationally recognised English language proficiency test is required.

The above undergraduate degrees have some minimal requirements on specializations which need to be fulfilled. In addition to the above required undergraduate degrees, students need to have 40 ECTS in basic undergraduate mathematics and programming courses (calculus, linear algebra and/or mathematical modeling and programming). A course in programming is compulsory. The average mark for the mathematics and programming courses, as well as 40 ECTS in senior undergraduate courses (2000 and 3000 level in Norway) for the specific specialization has at least to be C (letter marks). As an example, an undergraduate degree in Chemistry has a minimal requirement on chemistry courses, typically amounting to at least 60 ECTS out of 180 ECTS for a bachelor's degree. The average mark on the 40 ECTS of selected senior undergraduate credits in chemistry and the 40 ECTS in mathematics and programming should at least be C,