ISCL 2011

Third International ALP/GULP School on Computational Logic

Bertinoro, Italy, April 10-15, 2011 (co-located with DALT School 2011)

http://lia.deis.unibo.it/confs/iscl

*** EARLY REGISTRATION DEADLINE: MARCH 10, 2011 ***

Computational Logic has many applications, including the **modeling of intelligent systems**, **verification of software**, and the support of systems for **solving computationally hard problems**. Moreover, being founded on mathematical logic, tools based on CL are themselves amenable to **safe optimization and verification techniques**.
ISCL 2011 builds on the success of 6 schools organized by **GULP**, the Italian Association for Logic Programming. GULP, founded in 1985, is a non-profit organization which is in charge of organizing the Italian Conference on Computational Logic. ISCL 2011 is the result of a partnership between GULP and **ALP**, the Association for Logic Programing. It aims at giving a comprehensive introduction to this exciting research domain and disseminate the results of research with a perspective on the future. The school will provide a rich programme of lectures on different aspects of CL, covering both the **theoretical framework** and relevant **practical perspectives, techniques and tools**. Each lecture will provide the basic notions of its topic before proceeding to more advanced issues.
The school will include activities dedicated to graduate students and final exams on request.

TOPICS & LECTURERS

- *Constraint Languages for Parameterized Verification: Bags, Words, Trees, and Graphs.*
  Lecturer: **Giorgio Delzanno**.
  Giorgio Delzanno is Associate Professor the University of Genoa. He has given many important contributions in automated verification, model checking, infinite-state systems, models for concurrent and biological systems. He has been the recipient of several research grants and international awards.

- *Description Logics.*
  Lecturer: **Enrico Franconi**.
  Enrico Franconi is the Director of the European Masters Program in Computational Logic at the Free University of Bozen-Bolzano, and Principal Investigator in many EU-funded actions, networks of excellence and large-scale projects on topics related to the semantic web, networked knowledge, business processes and integration of ontological and rule-based reasoning.

- *Computational Logic and Human Thinking: How to be Artificially Intelligent.*
  Lecturer: **Robert Kowalski**.
  Robert Kowalski is Professor Emeritus at Imperial College London, and one of the first developers of logic programming. He made important contributions to various areas such as automated reasoning, representing and reasoning about time, abductive logic programming and intelligent agents. His current research focuses on the application of computational logic to cognitive science.

- *Unity in Computational Logic.*
  Lecturer: **Dale Miller**.
  Dale Miller is the Director of Research at INRIA Saclay and leader of the Parsifal team working on foundational aspects of proof theory as well as on the design and implementation of systems that exploit that foundational work. His main interests are in programming language theory, proof theory, linear logic, and automated deduction.

- *Constraint Programming and Optimization Systems.*
  Lecturer: **Pascal Van Hentenryck**.
  Pascal Van Hentenryck is Professor of computer science at Brown University, and the Director of the optimization laboratory. He was the main designer and implementor of the CHIP programming system. He leads many research projects funded by public and private institutions, in which his research is applied to a large number of domains.

TARGET AUDIENCE The school targets graduate students as well as other interested researchers, from university, government and industry. It will allow students to get a thorough overview of cutting-edge research and technologies and get in touch with leading scientists.
The school aims to be truly international with a strong participation from regions all around the world. This will help students make connections with international participants and set the base for potentially long-term cooperations.
An initial list of participants is available on the school Web site.

FINANCIAL AID AND MORE Grant application is now closed. However, additional support is still available to AEPIA, APPIA, ACIA, AIxIA and AISB members.
Limited personal subscription to selected journals will be offered by John Wiley & Sons to all attendees registered before March 25, 2011.

VENUE The University Residential Center is located in the small medieval hilltop town of Bertinoro, 50km east of Bologna at an elevation of 230m above sea level. Bertinoro is easily reachable from Bologna and Forli airport or train station. The registration includes shuttle bus on April 10 and April 15. Bertinoro is close to many splendid Italian locations such as Ravenna, Rimini on the Adriatic coast, and the Republic of San Marino (all within 35km). Bertinoro can also be a base for visiting some of the better-known Italian locations such as Padua, Ferrara, Venice, Urbino, Florence and Siena.

SPONSORS AI Journal, Association for Logic Programming, Italian Association for Logic Programming, Spanish Association for AI, Catalan Association for AI, Portuguese Association for AI, The British Society for the Study of AI and Simulation of Behaviour, Italian Association for AI, Italian Association for Logic Programming, SICStus Prolog, John Wiley & Sons, Bertinoro International Center for Informatics.

INQUIRIES Send your inquires to `iscl.2011@gmail.com`. We will answer in 2 working days.

ORGANISATION

School Organisers

Paolo Torroni, DEIS, University of Bologna, Italy
Maurizio Gabbrielli, DSI, University of Bologna, Italy

Student Session Organiser

Marco Montali, DEIS, University of Bologna, Italy

Local Organisers

Marco Prandini, DEIS, University of Bologna
Eleonora Campori, Bertinoro Center for Informatics
Manuela Schiavi, Bertinoro Center for Informatics

LECTURES

Constraint Languages for Parametrized Verification: Bags, Words, Trees, and Graphs

Parametrized verification is aimed at developing methods for proving the correctness of systems consisting of an arbitrary number of repeated components. In the lectures we overview some of the methods that can be applied to systems in which configurations can be represented by structures like bags, words, trees, and graphs. Examples of this class of systems are: *broadcast protocols* (used to model cache coherence protocols), *automata with global conditions* (used to model mutual exclusion protocols for N-processes), *tree rewriting systems* (used to model hierarchical systems), *selective broadcast protocols* (used to model protocols for ad hoc networks). In the presentation we use the metaphor "constraints as symbolic representation of sets of states" to give a uniform presentation of verification methods and of termination conditions in all these types of systems. Prerequisites: Basics of logic and algorithms. (Lecturer: **Giorgio Delzanno**).

Description Logics

The main effort of the research in knowledge representation is providing theories and systems for expressing structured knowledge and for accessing and reasoning with it in a principled way. In this course we will study Description Logics (DL), an important powerful class of logic-based knowledge representation languages, which also form the logical underpinning of the OWL family of web ontology languages standardised by W3C. The emphasis will be on a rigorous approach to knowledge representation and building ontologies. DL will be introduced with its simplest formalization; the computational properties and algorithms of the so called structural DL will be analysed. Then, the course considers propositional DL: we will study the computational properties and the reasoning with tableaux calculus. In the final part of the course, we will consider advanced topics such as the representation of knowledge bases and ontologies, and the connections of DL with database theory. (Lecturer: **Enrico Franconi**).

Computational Logic and Human Thinking: How to be Artificially Intelligent

This course is based on the book Computational Logic and Human Thinking: How to be Artificially Intelligent to be published by Cambridge University Press. In both this course and the book, I make the case for a comprehensive, logic-based theory of human intelligence, drawing upon and reconciling a number of otherwise competing paradigms in Artificial Intelligence and other fields. The most important of these paradigms are production systems, logic programming, classical logic and decision theory. The technical foundations of the theory are provided by abductive logic programming embedded in an observation-thought-decision-action agent cycle. The theory draws support, not only from Logic, Computing and Artificial Intelligence, but from related developments in Cognitive Psychology, Philosophy, Law and Management Science. (Lecturer: **Robert Kowalski**).

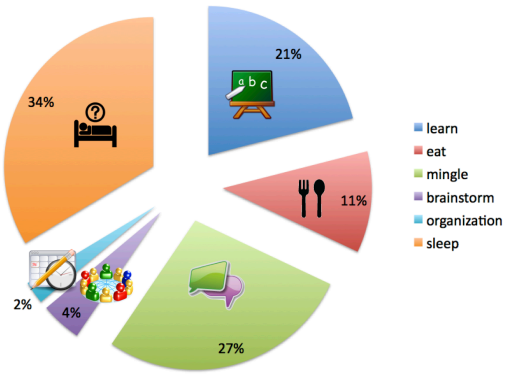
Unity in Computational Logic

Computational logic is divided into several different fragments. There is the division between the proof-as-program (functional programming) approach and the proof-search (logic programming) approach to specifying computation. There is the division among computation, model checking, and theorem proving. Even at the level of the description of such technical devices as proofs systems, there is the division among sequent calculus, natural deduction, tableaux, and resolution. In these lectures, I will show how recent results in structural proof theory bring an organization to these topics so that these divisions can be understood as certain choices within a large, flexible framework. That framework involves recent lessons learned from linear logic, focused proofs systems, and the use of fixed points and equality as logical connectives. (Lecturer: **Dale Miller**).

Constraint Programming and Optimization Systems

Constraint programming is a declarative paradigm for expressing and solving hard combinatorial optimization problems. Constraint programming features an expressive and compositional language for expressing constraints, which captures substructures on an application. Moreover, constraint programming typically offers a rich search language to guide the solver towards feasible and infeasible solutions. Computationally, constraint programming uses constraints to filter infeasible values from the variable domains. This course reviews both of these aspects, explores the hybridization of constraint programming with other optimization paradigms, and discusses similarities and differences with other approaches to optimization and constraint satisfaction. Real case studies in a modern constraint programming languages demonstrate the technology. (Lecturer: **Pascal Van Hentenryck**).

ACTIVITIES



The programme will include:

- an introductory lecture to give an overview of the school
- 5 topical courses of 6 hours each
- student sessions with focussed brainstorming and organized mentoring activities
- a social trip

The school will organize student examinations, on demand. For Italian students, the participation to all courses and successful result of the examination will correspond to 2 CFU.