Crossroads in Constraint Programming

Marius Silaghi, Christian Bessiere

To cite this version:

HAL Id: lirmm-00748664
https://hal-lirmm.ccsd.cnrs.fr/lirmm-00748664
Submitted on 5 Nov 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Crossroads in Constraint Programming

MARIUS SILAGHI, FLORIDA TECH, USA, marius.silaghi@fit.edu
CHRISTIAN BESSIERE, UNIVERSITY OF MONTPELIER, FRANCE, bessiere@lirmm.fr

I. CONSTRAINT PROGRAMMING

Constraint programming is a declarative programming paradigm exploiting techniques stemming from research on combinatorial problems in computer vision, and robot planning. The paradigm comes close to the dream that users only need to simply state a problem and the computer will solve it, as underlined in the seminal 1996 article In Pursuit of the Holy Grail by Eugene Freuder. The idea is that programming should be possible by simply stating the problem using a set of constraints. Using the traits of these constraints, appropriate techniques are then automatically selected and applied for solving the problem. Constraints are ubiquitous, and the requirement to satisfy them can be modeled within the framework of the constraint satisfaction problem (CSP). The constraints of a CSP are specified as relations on a set of variables. The choice of these variables and formulation of the constraints turns out to be essential for the efficiency of the obtained program. Many optimization problems can also be addressed with various extensions of constraint programming. Constraints whose satisfaction is optional are called soft constraints. They can be associated with a function that quantifies the desire for their satisfaction.

The brute force approach to combinatorial problems is usually considered to be one of either the chronological backtracking or the generate and test method. The research into constraint programming has started with work on local reasoning. Local reasoning combines sub-problems during backtracking. Techniques where all applied operations result only in redundant constraints or in the splitting of the search space guarantee that no solution is lost. Those techniques guaranteeing to find a solution whenever a solution exists are said to be complete. Most past research has focused on complete algorithms, and this is being identified as something that may change in the near future.

The propagation process is also referred to as local consistency enforcement since new constraints illustrate clearer what values are allowed. In general, the generation of unary constraints, i.e., removing values from domains, which has lower overhead, has been more successful in solvers, specially when repeatedly applied on subproblems during backtracking. Techniques where all applied operations result only in redundant constraints or in the splitting of the search space guarantee that no solution is lost. Those techniques guaranteeing to find a solution whenever a solution exists are said to be complete. Most past research has focused on complete algorithms, and this is being identified as something that may change in the near future.

II. FREUDERFEST AND CP 2011

The 17th International Conference on Principles and Practice of Constraint Programming was held in September 2011 in Perugia, Italy. Freuderfest, a special workshop dedicated to the retirement of Eugene Freuder, the founder and director of the Constraint Computation Center in Cork was held the day before the conference in the historical building of the administration of Perugia. As it was highlighted there by Francesca Rossi, Eugene has worked with 111 collaborators on 112 articles. Some of his closest collaborators were invited to speak about the impact of
his work on their research. The workshop was opened with an introduction by Barry O’Sullivan, the new director of 4C. Eugene himself used his speech to argue that the CP research community should debate more on the relevance of completeness and encouraged the intensification of research into incomplete techniques. This is a significant turn from his original focus on complete backtracking-free search and tree structures, interchangeability and symmetries. The other speakers focused on his seminal works in each of these fields, in which he is known to have launched and structured the basic ideas.

Eight collocated workshops were scheduled during the first day of the conference. Among them were the 11th Workshop on Soft Constraints (Soft), the 8th Workshop on Local Search (LSCS), 11th Workshop on Symmetry (SymCon), and 10th Workshop on Modeling and Reformulation (ModRef). There were also new workshops on CSP based Bioinformatics (WCB), Component Configuration (Lococo), Parallel Methods (PMCS), and on the MiniZinc Modeling language. The ModRef workshop organized a very popular panel on the available extensions to the common CSP representations. The discussions addressed the difficulties of extensions with Boolean Clauses, LP, MDDs and Neural Networks.

The main conference was started with an invited talk by Jean-Charles Régis on common pitfalls to avoid when solving problems with CP. His conclusion was that when CP does not work, most likely it is due to the choice of a bad representation. The Best Student Paper Award was offered to Marie Pelleau for her work on representing constraints in continuous domains using two Cartesian coordinate systems, significantly improving the approximation of complex shapes. Meinoff Sellman from IBM gave a tutorial on solver portfolios. He recommended SATzilla and CP Hydra as the best portfolio solvers for SAT and CSPs.

The Best Paper Award was offered to Georg Gottlob for proving an old conjecture about the NP-hardness of Minimal Constraint Networks. He shows that even when a constraint problem is reduced to its minimal configuration, namely where each pair of values for any two variables that is not forbidden by any constraint appears in at least one solution, finding a solution is a hard problem.

In a premiere for CP, the invited talk by Patrick Prosser with the occasion of receiving the Research Excellence Award from the Association of Constraint Programming (ACP) was delivered using a much appreciated YouTube video. The ACP Doctoral Research Award was offered to Standa Živný, who gave a talk on the complexity and expressive power of valued CSPs. He showed that many tractable constraint languages can be obtained by composing constraints from existing languages. The doctoral students tutorial by Laurent Michel focused on the importance of careful design for experiments, factoring out the noise and caching issues with sufficiently large problem sets.

Laurent Perron from Google described a successful application of constraint programming within the most important Google servers, namely for taking caching decisions and decisions concerning the routing of user traffic. The constraint programming technique has saved 1 millisecond which translates into savings of millions of dollars given the scale of the corresponding operations at Google. Google proposed the ROADEF challenge which stresses the need for a good solution fast (no optimality and no completeness required). The challenge has deadlines in each August and February. The Best Application Paper Award was offered for a paper coauthored by Venkatesh Ramamoorthy for successfully using CSPs to design highly nonlinear cryptographic functions. Such functions are the most sensible part of modern ciphers, and resistance to known attacks requires the satisfaction of a set of constraints. No solution is known to satisfy all these constraints, but some of them are soft and solutions have been found within certain thresholds. A few of these constraints are global and their decomposition, when done without preserving completeness, enabled to find functions that better satisfy the soft constraints when compared to previously found ones.

The conference was closed with a celebration of Gene Freuder’s retirement, under the form of a panel on the future of constraint programming. Eugene stressed the importance of selecting lofty goals, launching buzzwords like: Problem solving Web, Electronic Embodiment, Computational Precognition and Business Constraints. Visible applications are in healthcare, analytics, energy, sustainability, and humanitarian operations. The researchers should balance the just-in-case redundant constraint generation tendency that challenges memory requirements for the just-in-time approach that spends resources only when needed. Approaching the lofty goals should be achieved also by broadening the scope of the CP conference which should join all research where the constraint is relevant, no longer focusing only on algorithms based on backtracking.

All the presentation slides were made available on website of the program of the conference: dmi.unipg.it/cp2011.