



Postdoctoral proposal in artificial intelligence and machine learning

Pushing the computational frontiers of reasoning with logic, probabilities and preferences

Advisors:

Simon de Givry, MIAT, INRA, simon.de-givry@inra.fr

Thomas Schiex, MIAT, INRA, thomas.schiex@inra.fr

Net salary: $\geq 2600\text{€}$ per month with some teaching (64 hours per year on average)

Duration: from 1 to 4 years, starting in autumn 2019

Location: Institut National de la Recherche en Agronomie (INRA)

Laboratoire de Mathématiques et Informatique Appliquées de Toulouse (MIAT)

24, Chemin de Borde-Rouge, Castanet-Tolosan (near Toulouse)

Constraint programming (CP [Rossi et al, 2006]) is an AI *Automated Reasoning* technology with tight connections with propositional logic. It offers a problem modeling and solving framework where the set of solutions of a complex (NP-hard) problem is described by discrete variables, connected by constraints (simple Boolean functions). Together with propositional satisfiability, it is one of the automated reasoning approaches of AI, where problems are solved *exactly* to provide rigorous solutions to hardware or software testing and verification, system configuration, scheduling or planning problems.

Discrete Stochastic Graphical Models (GMs [Koller et al, 2009]) define a *Machine Learning* technology where a probability mass function is described by discrete variables, connected by potentials (simple numerical functions). GMs can be learned from data and the NP-hard problem of identifying a Maximum a Posteriori (MAP) labelling is often solved *approximately* to tackle several problems in Image [Kappes et al, 2013] and Natural Language Processing [Bilmes, 2004], among others.

The Cost Function Network framework [Cooper et al, 2010] with its associated C++ open source award-winning solver [toulbar2](#), developed in our team, combine the ideas of Constraint Programming and Stochastic Graphical Models. By solving the so-called Weighted Constraint Satisfaction problem, [toulbar2](#) is capable of simultaneously reasoning on logical information described as Boolean functions and gradual, possibly Machine Learned, information described as local numerical functions.

To process the available information, the solver relies on a guaranteed Hybrid Branch and Bound-based algorithm (HBFS) that combines Best-First and Depth-First search [Allouche et al, 2015]. In this algorithm, pruning follows from a variety of mechanisms that can either simplify the problem at hand, provide primal solutions (using local search, rounding or incomplete tree-search), or provide dual solutions and lower bounds. Multi-threading and parallel solving offer new opportunities to organize these various mechanisms differently in time, to exploit problem decompositions, to apply stronger primal/dual reasoning and use

Machine learning to guide search or decide which mechanism to activate based on the current solving and/or a collection of instances of the same problem.

The short-term aim of the PostDoc will be to enhance the current solver with parallel solving capabilities on multi-core, cluster, and cloud architectures following the ongoing trend that is followed by CP, SAT, and also Integer Linear Programming solvers [Gent et al, 2018 ; Hamadi et al, 2018]. A first master-worker parallel version of the HBFS algorithm has been implemented [Beldjilali et al, 2019]. The PostDoc will explore new machine learning rules for better load balancing and reduced workload of the master process. He or she will continue to adapt the parallel HBFS algorithm by exploiting tree decompositions [Jégou et al, 2017]. The resulting algorithm will be integrated into a metaheuristic [Ouali et al, 2017] for improved global anytime lower and upper bounds.

Experiments will be performed on large collections of real problem instances, many of which are not known to be currently solvable. This includes the possible application of toulbar2 onto current exciting problems in Computational Protein Design (CPD) [Allouche et al, 2014 ; Pan et al, 2016], in collaboration with molecular modellers and biochemists, and in the context of the ongoing development of a dedicated CPD software.

The position is specifically open to highly creative researchers that may quickly want to develop and explore their own ideas. As such, we expect that the PostDoc will be increasingly capable of injecting their own ideas in the project, in interaction with all the members of the project team as well as external collaborators, and contribute to the supervision of PhD students.

Candidate profile

The PostDoc is at the intersection of CP, SAT, integer programming, metaheuristics, and distributed computing. The ideal candidate should therefore be familiar with CP or SAT algorithms. He or she may also benefit from background knowledge in the weighted variants of SAT/CP, in Integer Linear Programming, or in Stochastic Graphical Models processing. Some experience in the design and implementation of multi-threaded/distributed code is a nice plus. Good programming abilities (in C++ ideally) will be required. Additional knowledge in bioinformatics, biochemistry, and molecular modelling would be a plus in the context of Computational Protein Design applications.

Send by email to the postdoc advisors your detailed CV, a motivation letter, and transcripts of bachelor's degree. Samples of published research by the candidate and reference letters will be a plus.

References

D Allouche, I André, S Barbe, J Davies, S de Givry, G Katsirelos, B O'Sullivan, S Prestwich, T Schiex, S Traoré. Computational protein design as an optimization problem. In Journal of Artificial Intelligence 212, 59-79, 2014.

D Allouche, S. de Givry, G. Katsirelos, T. Schiex, M. Zytnicki. Anytime Hybrid Best-First Search with Tree Decomposition for Weighted CSP. In Proc. of CP-15, pp 12-28, Cork, Ireland, 2015.

A Beldjilali, D Allouche, S de Givry. Parallel Hybrid Best-First Search. Poster at CP-19, Stamford, CT, 2019.

J Bilmes. Graphical Models and Automatic Speech Recognition. Mathematical Foundations of Speech and Language Processing pp 191-245, 2004.

M Cooper, S. de Givry, M. Sanchez, T. Schiex, M. Zytnicky, T. Werner. Soft Arc-consistency revisited. In Journal of Artificial Intelligence, 174(7-8), pp 449-478, 2010.

I Gent, I Miguel, P. Nightingale, C. McCreesh, P. Prosser, N. Moore, and C. Unsworth. A review of literature on parallel constraint solving. In Journal of Theory and Practice of Logic Programming, 18(5-6), 725-758, 2018.

Y. Hamadi and Lakhdar Sais. Handbook of Parallel Constraint Reasoning. Springer, 2018.

P Jégou, H Kanso, and C Terrioux. Adaptive and Opportunistic Exploitation of Tree-Decompositions for Weighted CSPs. In Proc. of ICTAI 2017, pp 366-373, Boston, MA, USA, 2017.

J Kappes, M Speth, G Reinelt, and C Schnorr. Towards efficient and exact MAP-inference for large scale discrete computer vision problems via combinatorial optimization. In Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, 2013.

D Koller, N Friedman. Probabilistic graphical models: principles and techniques. The MIT Press, 2009.

A Ouali, D Allouche, S de Givry, S Loudni, Y Lebbah, F Eckhardt, and L Loukil. Iterative Decomposition Guided Variable Neighborhood Search for Graphical Model Energy Minimization. In Proc. of UAI-17, pp 550-559, Sydney, Australia, 2017.

Y Pan, Y Dong, J Zhou, M Hallen, B Donald, J Zeng, and W Xu. cOSPNEY: A Cloud-Based Distributed Algorithm for Large-Scale Computational Protein Design. In Journal of Computational Biology, 23(9), 2016.

F Rossi, P Van Beek, and T Walsh. Handbook of constraint programming. Elsevier, 2006.