Industrial Success Stories of ASP and CP: What's still open?

Gerhard Friedrich
Institut für Angewandte Informatik
Alpen-Adria Universität Klagenfurt, Austria
(*e-mail:* gerhard.friedrich@aau.at)

Abstract

More than 25 years ago together with Siemens we started to investigate the possibility of substituting the classical rule-based configuration approach by model-based techniques. It turned out that in those days only constrained programming (CP) had any real chance of meeting the application demands. By exploiting CP we were able to significantly improve the productivity of highly trained employees (by more than 300%) and to substantially reduce software development and maintenance costs (by more than 80%) (FFH+98). Consequently, CP has been our method of choice for problem solving in industrial projects since 1989.

Some years ago, we started to investigate answer set programming (ASP) techniques (BET11), mainly because of the possibility to apply a very expressive logical first-order language for specifying problems. It emerged that, by using simply problem encoding, we were able to solve difficult real world problem instances witnessing the enormous improvements of logic programming over the last decades (ADF+11).

Although ASP and CP have proven their practical applicability, we will point out challenges of large problems of the electronic and the semiconductor industry. In particular, we will stress the power of problem-specific heuristics (GKR+13; STW+13) which turned out to be the key in many applications of problem solvers.

Looking at the famous equation "algorithm = logic + control" (Kow79) most of the current work in the AI community assumes that control should be problem-independent and only the logical specification depends on the problem to be solved, i.e. "algorithm = logic(problem) + control". It is not surprising that for the current problem solving technology this is a practical approach up to a certain size of the problem instances, since we deal with NP-hard problems in many cases. However, it is observed (and examples are given (TFF12; MF15)) that problem-specific heuristics allow enormous run-time improvements. This success is based on problem-specific control, i.e. "algorithm = logic(problem) + *control(problem)*". Unfortunately, the design of such problem-specific heuristics is very time-consuming and redesigns are frequently required because of recurrent changes of the problem. Interestingly, humans are very successful at developing such problem-specific heuristics. Therefore, we argue that the automation of generating problem-specific heuristics with satisfying quality is still an important basic AI research goal with high practical impact that should be achievable (Pea83).

References

Markus Aschinger, Conrad Drescher, Gerhard Friedrich, Georg Gottlob, Peter Jeavons, Anna Ryabokon, and Evgenij Thorstensen. Optimization methods for the partner units problem. In Tobias Achterberg and J. Christopher Beck, editors, *Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems - 8th International Conference, CPAIOR 2011, Berlin, Germany, May 23-27, 2011. Proceedings*, volume 6697 of *Lecture Notes in Computer Science*, pages 4–19. Springer, 2011.

Gerhard Brewka, Thomas Eiter, and Miroslaw Truszczynski. Answer set programming at a glance. *Commun. ACM*, 54(12):92–103, 2011.

Gerhard Fleischanderl, Gerhard Friedrich, Alois Haselböck, Herwig Schreiner, and Markus Stumptner. Configuring large systems using generative constraint satisfaction. *IEEE Intelligent Systems*, 13(4):59–68, 1998.

Martin Gebser, Benjamin Kaufmann, Javier Romero, Ramón Otero, Torsten Schaub, and Philipp Wanko. Domain-specific heuristics in answer set programming. In Marie desJardins and Michael L. Littman, editors, *Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence, July 14-18, 2013, Bellevue, Washington, USA*. AAAI Press, 2013.

Robert A. Kowalski. Algorithm = logic + control. *Commun. ACM*, 22(7):424–436, 1979.

Vera Mersheeva and Gerhard Friedrich. Multi-uav monitoring with priorities and limited energy resources. In Ronen I. Brafman, Carmel Domshlak, Patrik Haslum, and Shlomo Zilberstein, editors, *Proceedings of the Twenty-Fifth International Conference on Automated Planning and Scheduling, ICAPS 2015, Jerusalem, Israel, June 7-11, 2015.*, pages 347–356. AAAI Press, 2015.

Judea Pearl. On the discovery and generation of certain heuristics. *AI Magazine*, 4(1):23–33, 1983.

Tom Schrijvers, Guido Tack, Pieter Wuille, Horst Samulowitz, and Peter J. Stuckey. Search combinators. *Constraints*, 18(2):269–305, 2013.

Erich Christian Teppan, Gerhard Friedrich, and Andreas A. Falkner. Quickpup: A heuristic backtracking algorithm for the partner units configuration problem. In Markus P. J. Fromherz and Hector Muñoz-Avila, editors, *Proceedings of the Twenty-Fourth Conference on Innovative Applications of Artificial Intelligence, July 22-26, 2012, Toronto, Ontario, Canada*. AAAI, 2012.