Further Experimental Analysis

Chuan Luo[1,2], Holger H. Hoos[2], and Shaowei Cai[3]

[1] Microsoft Research, China
[2] Leiden Institute of Advanced Computer Science, Leiden University, The Netherlands
[3] State Key Laboratory of Computer Science, Institute of Software, Chinese Academy of Sciences, China
chuan.luo@microsoft.com, hh@liacs.nl, caisw@ios.ac.cn

1 The effect of automatically configuring *PbO-CCSAT*

We recall that the default configuration of *PbO-CCSAT* is equivalent to *DCCASat* (the version used for solving structured SAT instances), which inspired much of the design of our CC-based SLS framework. As seen from the results presented so far, automatically configuring the flexible *PbO-CCSAT* framework leads to substantial improvements of the average performance on most of our benchmark sets, up to a factor of over 1000 in terms of PAR10 score for `PTN [Test]`. To further investigate these performance gains, it is instructive to look at them on a per-instance basis. As seen in Figures 1–4, configuration leads to performance improvements on the large majority of instances for 6 of our 7 satisfiable benchmark sets, including the prominent and challenging `FCC`, `PTN` and `SC17-mp1-9` instances.

The remaining set, `SMT-QF-BV [Test]`, is quite challenging, and performance improvements of more than 2 orders of magnitude are achieved for many instances; still, on a significant number of instances, optimisation of aggregate performance through automated configuration leads to performance worse than the default version of *PbO-CCSAT*. Since automated configuration optimises average performance (to be precise, PAR10), this kind of trade-off is not unexpected, and it is, in fact, surprising that performance improvement on the remaining, challenging benchmark set, is observed for the vast majority of instances.

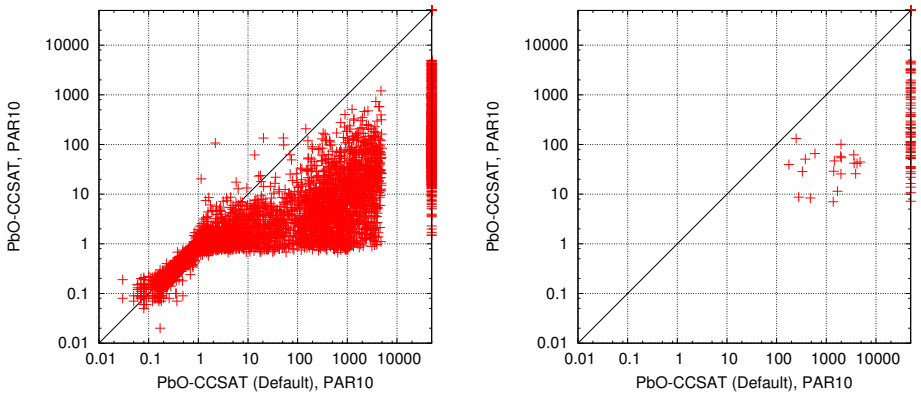

Fig. 1: Scatter plots corresponding to the performance comparison between *PbO-CCSAT* and *PbO-CCSAT (Default)* on `FCC-SAT [Test]` (left) and `FCC-UNKNOWN` (right).

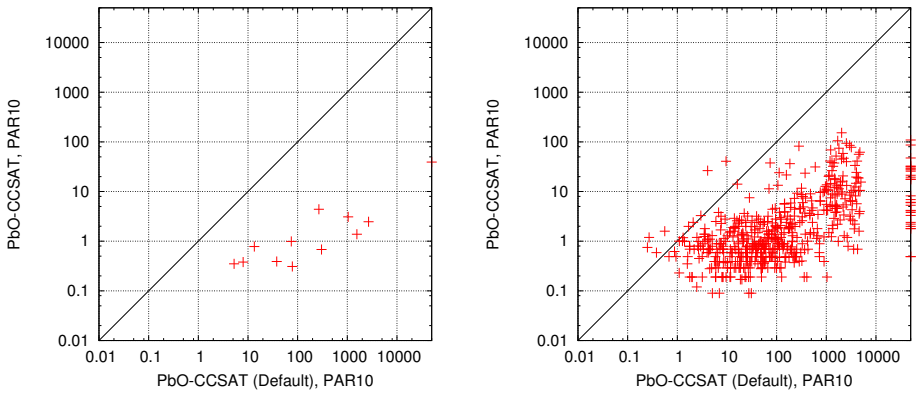

Fig. 2: Scatter plots corresponding to the performance comparison between *PbO-CCSAT* and *PbO-CCSAT (Default)* on `PTN [Test]` (left) and `PTN-More` (right).

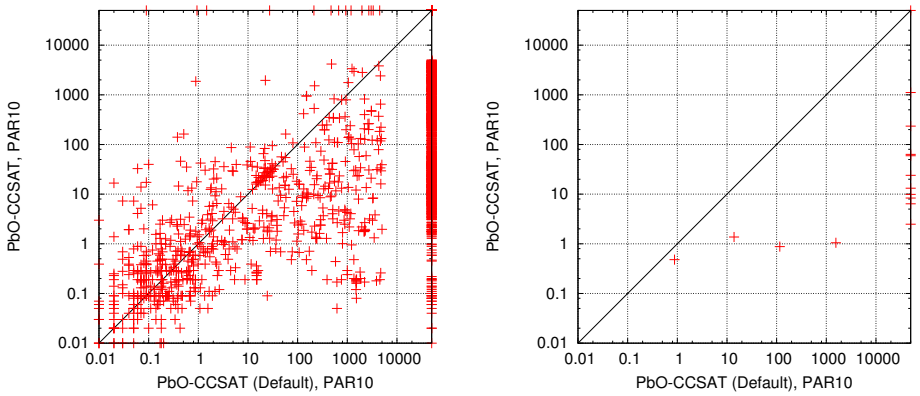

Fig. 3: Scatter plots corresponding to the performance comparison between *PbO-CCSAT* and *PbO-CCSAT (Default)* on `SMT-QF-BV [Test]` (left) and `Community [Test]` (right).

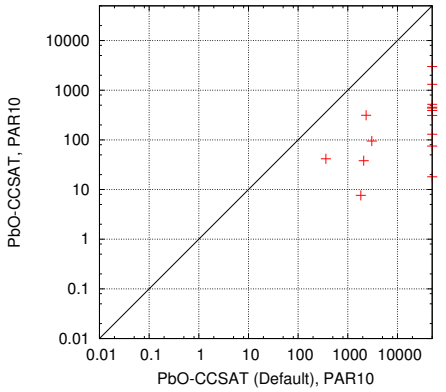

Fig. 4: Scatter plot corresponding to the performance comparison between *PbO-CCSAT* and *PbO-CCSAT (Default)* on `SC17-mp1-9 [Test]`.