

Heavy-tailed Runtime Distributions: Heuristics, Models and Optimal Refutations*

Tudor Hulubei and Barry O’Sullivan

Cork Constraint Computation Centre
Department of Computer Science, University College Cork, Ireland
tudor@hulubei.net, b.osullivan@cs.ucc.ie

1 Introduction

We perform an in-depth empirical study of runtime distributions associated with a continuum of problem formulations for QWH-10 with 90% holes¹ defined between the points where a formulation is entirely specified in terms of binary inequality constraints to models specified using an increasing number of ALLDIFFERENT global constraints [4]. For each model we study a variety of variable and value ordering heuristics. We compare their runtime distributions against runtime distributions where any mistakes made in search are refuted optimally [2], and make the following observations:

1. For the problems considered, *variations in the heuristics used have a far more significant effect on hybrid models* (i.e. models using both binary and global constraints) than they do on purely binary models.
2. While algorithms tend to perform better on hybrid models, a straight line can still be observed in a log-log plot of their runtime distributions, even when mistakes are refuted optimally. In other words, *runtime distributions of hybrid models can remain inherently heavy-tailed* [3].
3. *Models using global constraints are not always better than purely binary models.* We encountered configurations where increasing the number of global constraints used to enforce distinct values on rows and columns (and removing the corresponding sets of binary constraints) does not lead to a monotonic decrease in search effort. *The discrepancy all but disappeared when we looked at the corresponding (quasi-)optimal refutations* for the exact same configuration.
4. With the exception of a few unusual cases, using global constraints did improve search performance and, when that occurred, *the refutations encountered for hybrid models were much closer to their corresponding optimal* than for the binary model.
5. For the problems considered, the variables used in refuting a mistake usually represent only a small fraction of the variables still uninstantiated at the time the mistake was made, *yet this small subset of variables can most of the time be re-ordered to refute the mistake optimally.*

* This work was supported by Science Foundation Ireland (Grant 00/PI.1/C075). We thank the Boole Centre For Research in Informatics for providing access to their Beowulf cluster.

¹ We choose under-constrained problems in order to study heavy-tailed runtime distributions [1].

2 Experiments

Our experiments were performed on satisfiable QWH-10 problem instances with 90% random balanced holes, and included 4 variable ordering heuristics: min-domain, min-dom/ddeg², breaz and min-dom/wdeg, and 3 value ordering heuristics: random, min-conflicts and max-conflicts, totalling 12 algorithms (we broke ties randomly). Most binary instances were too difficult to solve using random variable orderings or variable ordering anti-heuristics, which is why these heuristics have not been included.

We began our study of various ways to model a CSP by encoding these problems using only binary constraints (propagated using MAC) and then gradually replacing the binary constraints used to enforce distinct cells on rows and columns with equivalent n-ary ALLDIFFERENT constraints (which propagate generalised arc-consistency). In addition to the binary model, we represented QWH-10 using 3 different hybrid models by randomly selecting 2, 4, and 8 (out of the 20 possible) rows and/or columns and replacing their corresponding binary constraints with a single n-ary ALLDIFFERENT constraint. We will use *hybrid = X* to denote a certain model, with *X* being 0 for the binary case and 2, 4, or 8 for the others (16 and 20 are too easy).

We refer to a *mistake point* [2] as an assignment that cannot lead to a solution even though one existed before that assignment was made. An *actual refutation* is the search tree corresponding to a mistake, as obtained by some algorithm, with the *optimal refutation* for that mistake corresponding to a search tree of minimum size. Finally, the *quasi-optimal refutation* is the smallest refutation whose height does not exceed that of the actual refutation.

3 Results

The plots in Figures 1(a) and 1(b) show the actual and (quasi-)optimal runtime distributions of our 12 algorithms on the binary and hybrid models. We use the term *shorter* to refer to refutations that are either optimal, quasi-optimal, or simply the shortest improved refutations we could find that were smaller than the corresponding actual refutations, and the term *restricted shorter* to denote the smallest refutations we could find when the search for optimal refutations was restricted to the variables involved in the actual refutation. We use the term *cumulative effort* to refer to the effort required to refute all the mistakes encountered in a given instance.

Figure 1(a) shows the runtime distributions of our algorithms on the binary and hybrid models. MAC+min-conflicts+min-dom/wdeg is the only algorithm that succeeds in eliminating heavy tails for the binary model [3] and keeps doing so as we add more global constraints, while all other algorithms remain heavy-tailed for all hybrid models (hybrid=*X* with $X \leq 8$). Moreover, Figure 1(b) shows that, for some algorithms, heavy tails do not disappear even when the mistakes encountered are refuted optimally. In other words, for almost all hybrid models studied, even if we were able to use an oracle to refute insoluble subtrees optimally, for some combinations of heuristics we would still see heavy tails (*inherent heavy-tailedness*).

² We abbreviate dynamic-degree as ‘ddeg’ and weighted-degree as ‘wdeg’.

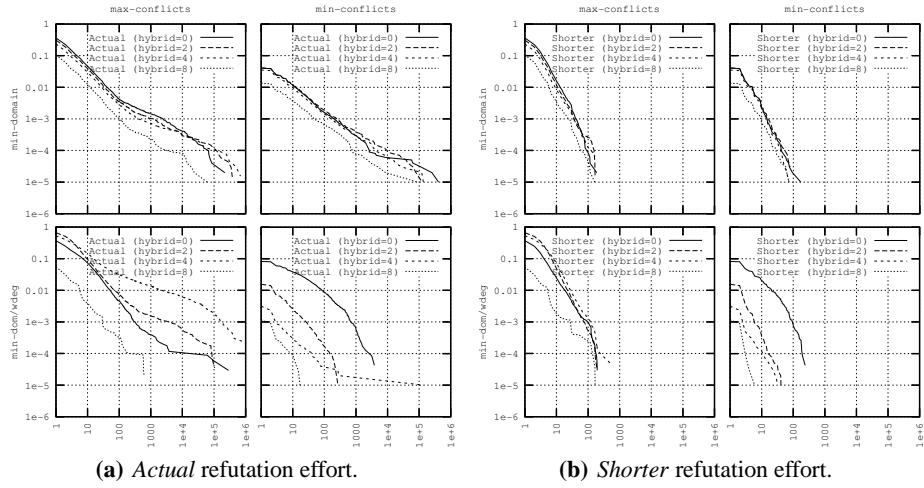


Fig. 1: Complement of the CDF (y-axis) of the actual (left) and shorter (right) effort (x-axis). We vary the value ordering across columns and variable ordering across rows.

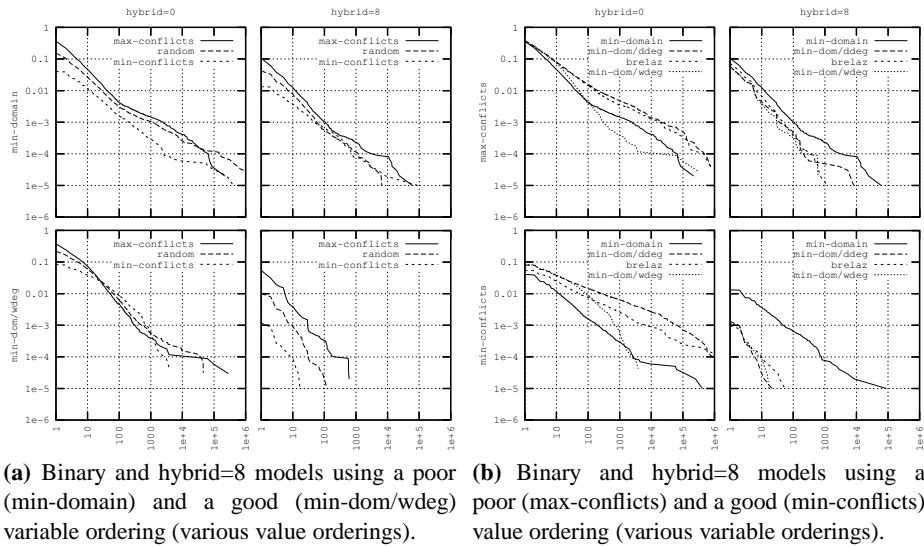


Fig. 2: Complement of the CDF (y-axis) of the cumulative *actual* effort (x-axis).

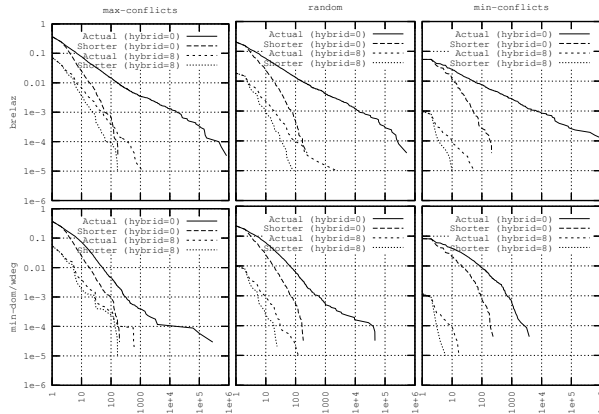


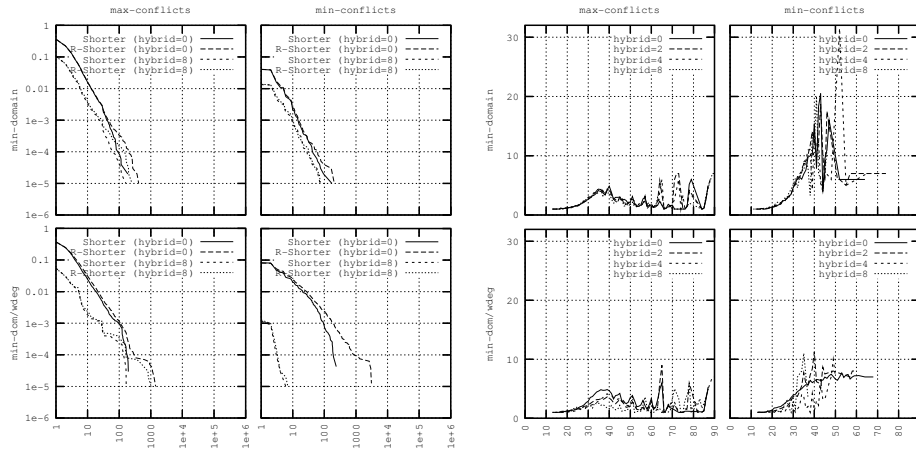
Fig. 3: Comparison of the complement of the CDF (y-axis) for the cumulative *actual* and *shorter* effort (x-axis) for the binary and hybrid=8 models.

Figures 1(a) and 1(b) also show that as our models become more sophisticated through the addition of global constraints, the slopes of the runtime distributions decrease to the point they can no longer be considered heavy-tailed. However, more sophisticated models do not always lead to better performance. Combining max-conflicts with min-dom/wdeg performs worse for the hybrid=4 model than for the binary model (hybrid=0), as can be seen in Figure 1(a). In the corresponding optimal refutations, while the binary model still outperforms the hybrid=4 model, it only does so by a very small margin. Also, it seems that more sophisticated models can benefit more from good ordering heuristics than the equivalent binary models.

Figure 2(a) shows that the performance improvements due to better value orderings heuristics become significant when a good variable ordering (min-dom/wdeg) is applied to the hybrid=8 model. Similarly, Figure 2(b) shows that the performance improvements due to better variable ordering heuristics become significant when a good value ordering heuristic (min-conflicts) is applied to the hybrid=8 model. These improvements are closely correlated with the runtime distribution of the (quasi-)optimal refutations (not shown for lack of space). Hardly any difference can be observed for the binary models. This analysis suggests that heuristics can infer more from more sophisticated models.

Figure 3 shows another advantage of using global constraints: compared with binary models, actual refutations are closer to optimality. Finally, Figure 4(a) shows that for the binary and hybrid=8 models, the (quasi-)optimal and restricted (quasi-)optimal refutations have almost identical runtime distributions³. Interestingly, the average number of variables involved in the actual refutations is only a small fraction of the total number of variables still uninstantiated when the mistake was made (Figure 4(b)). These observations suggest that all the variable ordering heuristics studied here select a very small

³ Due to time and technical constraints, we ran the *quasi-optimal* experiment and *restricted* experiments in parallel, generating potentially different instances, which explains why occasionally the restricted effort appears to be less than the quasi-optimal.



(a) Comparison of the complement of the CDF (y-axis) for the cumulative *shorter* and *restricted shorter* (*R-Shorter* in the plots) (x-axis) refutations for the binary and hybrid=8 models. (b) Average number of variables involved in the *actual* refutations (y-axis) as a function of the number of variables still uninstantiated at the time a mistake was made (x-axis).

Fig. 4: Statistics on restricted refutations.

subset of the remaining uninstantiated variables that could be re-ordered to obtain an optimal refutation. What differentiates a good heuristic from a poor one is the ability to select those variables in an order that minimises the size of the refutation.

4 Conclusions

We have shown empirically that for QWH-10, variations in heuristics have a greater effect on formulations involving a mix of binary and global constraints than on purely binary models. Models using global constraints are not always better than purely binary models. We have also shown that the small subset of variables used by a heuristic to refute a mistake can be *re-ordered* to obtain an almost optimal refutation. This raises the question of why heuristics select the right variables, but fail to find better refutations.

References

1. C.P. Gomes, C. Fernández, B. Selman, and C. Bessière. Statistical regimes across constrainedness regions. *Constraints*, 10(4):317–337, 2005.
2. T. Hulubei and B. O’Sullivan. Optimal refutations for constraint satisfaction problems. In *Proceedings of IJCAI-2005*, pages 163–168, 2005.
3. T. Hulubei and B. O’Sullivan. The impact of search heuristics on heavy-tailed behaviour. *Constraints*, 11(2–3):157–176, 2006.
4. J. C. Regin. A filtering algorithm for constraints of difference in CSPs. In *Proceedings of AAAI*, pages 362–367, 1994.