

Discovering Archipelagos of Tractability: Split-Backdoors to Constraint Satisfaction^{*}

Abstract

Robert Ganian¹, M.S. Ramanujan², and Stefan Szeider¹

¹ Vienna University of Technology, Vienna, Austria
rganian@gmail.com, stefan@szeider.net

² University of Bergen, Bergen, Norway
ramanujan.sridharan@ii.uib.no

The Constraint Satisfaction Problem (CSP) is a central and generic computational problem which provides a common framework for many theoretical and practical applications. A central line of research is concerned with the identification of classes of instances for which the CSP can be solved in polynomial time, such classes are often called “islands of tractability.” A prominent way of defining islands of tractability for the CSP is to restrict the relations that may occur in the constraints to a fixed set, called a *constraint language*. Schaefer’s famous Dichotomy Theorem [3] identifies all islands of tractability in terms of tractable constraint languages over a Boolean domain of values. Since then many extensions and generalizations of this result have been obtained.

Our results build upon these results and show how the identified islands of tractability can be combined to “archipelagos of tractability” (in general, the union of tractable constraint languages is not tractable). In order to put this to work, we need some control over the way how the constraints over the various considered tractable languages interact with each other. We capture this by the notion of a *split-backdoor* which is a set of variables that, when instantiated, separate the CSP instance into pairwise independent parts, each of which falls into a different island of tractability. Strong backdoors, as introduced by Williams et al. [4], form the special case where all instantiations of the backdoor variables move the instance into one fixed island.

Here the main difficulty lies in finding a split-backdoor of given size k . Indeed, once the split-backdoor is found, we can try all possible instantiations of the backdoor variables and apply the polynomial time algorithms associated with the islands of tractability. Our main result is an algorithm that finds a split-backdoor into a list of finite tractable (or $\#$ -tractable)

^{*} Research supported by the European Research Council (COMPLEX REASON, 239962) and the FWF Austrian Science Fund (X-TRACT, P26696).

constraint languages which contain all unary constant constraints; the algorithm runs in time $2^{\mathcal{O}(k^4)}n^{\mathcal{O}(1)}$ where k is the size of the split-backdoor. This also gives the running time for solving the decision or counting version of CSP for the given instance. We remark that as a special case, our algorithm can build upon Bulatov’s recent full characterization of all islands of tractability for CSP and #CSP (the counting version of CSP) that are defined in terms of so-called conservative constraint languages [1, 2].

As part of our algorithm for finding the split-backdoors, we first develop an algorithm to compute “split-modulators” in a graph theoretical setting. This has the advantage that we can build upon existing advanced algorithmic techniques. In particular, we show that for any finite collection of finite sets of (possibly disconnected) graphs $\mathcal{F}$, there is an algorithm that, given a graph G and a positive integer k , in time $2^{\mathcal{O}(k^4)}n^{\mathcal{O}(1)}$ either finds a set $X \subseteq V(G)$ of size at most k such that for each connected component C of $G - X$ there is some $\mathcal{F}_i \in \mathcal{F}$ such that C has no graph from $\mathcal{F}_i$ as an induced subgraph, or correctly reports that no such set X exists. This result has numerous applications outside CSP, as it can be used to combine known polynomial time algorithms for NP-hard problems restricted to classes of instances characterized by forbidden induced subgraphs.

References

1. A. A. Bulatov. Complexity of conservative constraint satisfaction problems. *ACM Trans. Comput. Log.*, 12(4):Art. 24, 66, 2011.
2. A. A. Bulatov. The complexity of the counting constraint satisfaction problem. *J. of the ACM*, 60(5):Art 34, 41, 2013.
3. T. J. Schaefer. The complexity of satisfiability problems. In *Conference Record of the Tenth Annual ACM Symposium on Theory of Computing (San Diego, Calif., 1978)*, pages 216–226. ACM, 1978.
4. R. Williams, C. Gomes, and B. Selman. Backdoors to typical case complexity. In G. Gottlob and T. Walsh, editors, *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence, IJCAI 2003*, pages 1173–1178. Morgan Kaufmann, 2003.