Native Cardinality Constraints: More Expressive, More Efficient Constraints

Jordyn C. Maglalang  
*Illinois Wesleyan University*

Mark Liffiton, Faculty Advisor  
*Illinois Wesleyan University*

Follow this and additional works at: [http://digitalcommons.iwu.edu/jwprc](http://digitalcommons.iwu.edu/jwprc)
Boolean cardinality constraints are commonly translated (encoded) into Boolean CNF clauses, a standard form of Boolean satisfiability, which can be solved using a standard SAT solving program. However, cardinality constraints are a simple generalization of clauses, and the complexity entailed by encoding them into CNF can be avoided by reasoning about cardinality constraints **natively** within a SAT solver. In this work, we compare the performance of two forms of native cardinality constraints against some of the best performing encodings from the literature. We designed a number of cardinality constraints including crafted, random and application problems, to be run in parallel on a cluster of computers. Native implementations substantially outperform CNF encodings on instances composed entirely of cardinality constraints, and instances that are mostly clauses with few cardinality constraints exhibit mixed results warranting further study.