Green-Tao Numbers and SAT

Oliver Kullmann

Computer Science Department
Swansea University
O.Kullmann@Swansea.ac.uk

http://cs.swan.ac.uk/~csoliver

Abstract. We aim at using the problems from exact Ramsey theory, concerned with computing Ramsey-type numbers, as a rich source of test problems for SAT solving, targeting especially hard problems. Particularly we consider the links between Ramsey theory in the integers, based on van der Waerden’s theorem, and (boolean, CNF) SAT solving. Based on Green-Tao’s theorem (“the primes contain arbitrarily long arithmetic progressions”) we introduce the Green-Tao numbers $\text{grt}_m(k_1, \ldots, k_m)$, which in a sense combine the strict structure of van der Waerden problems with the quasi-randomness of the distribution of prime numbers. In general the problem sizes become quickly infeasible here, but we show that for transversal extensions these numbers only grow linearly, thus having a method at hand to produce more problem instances of feasible sizes. Using standard SAT solvers (look-ahead, conflict-driven, and local search) we determine the basic Green-Tao numbers. It turns out that already for this single case of a Ramsey-type problem, when considering the best-performing solvers a wide variety of solver types is covered. This is different to van der Waerden problems, where apparently only simple look-ahead solvers succeed (regarding complete methods). For $m > 2$ the problems are non-boolean, and we introduce the generic translation scheme for translating non-boolean clause-sets to boolean clause-set. This general method offers an infinite variety of translations (“encodings”) and covers the known methods. In most cases the special instance called nested translation proved to be far superior over its competitors (including the direct translation).

1 Introduction

The applicability of SAT solvers has made tremendous progress over the last 15 years; see the recent handbook [3]. We are concerned here with solving (concrete) combinatorial problems (see [24] for an overview). Especially we are concerned with the computation of van-der-Waerden-like numbers, which is about colouring hypergraphs of arithmetic progressions; see [15] for the underlying report.

An arithmetic progression of size $k \in \mathbb{N}_0$ in $\mathbb{N}$ is a set $P \subset \mathbb{N}$ of size $k$ such that after ordering (in the natural order), two neighbours always have the same distance. So the arithmetic progressions of size $k > 1$ are the sets of the form $P = \{a + i \cdot d : i \in \{0, \ldots, k-1\}\}$ for $a, d \in \mathbb{N}$. Van der Waerden’s Theorem ([23])
shows that for every progression size \( k \in \mathbb{N} \) and every number \( m \in \mathbb{N} \) of parts there exists some \( n_0 \in \mathbb{N} \) such that for \( n \geq n_0 \), every partitioning of \( \{1, \ldots, n\} \) into \( m \) parts has some part which contains an arithmetic progression of size \( k \).

The smallest such \( n_0 \) is denoted by \( \text{vdw}_m(k) \), and is called a \( \text{vdW}-\text{number} \).

The subfield of Ramsey theory concerned with van der Waerden’s theorem is for over 70 years now an active field of mathematics and combinatorics; for an elementary introduction see [19].

We are concerned here with exact Ramsey theory, that is, computing \( \text{vdW} \)-like numbers if possible, or otherwise producing (concrete) lower bounds. [5] introduced the application of SAT for computing \( \text{vdW} \)-numbers, showing that all known \( \text{vdW} \)-numbers (at that time) were rather easily computable with SAT solvers. With [11] yet SAT had its biggest success, computing the new (major) \( \text{vdW} \)-number \( \text{vdw}_2(6) = 1132 \) (mentioned in [19] as a difficult research problem). See [12,18] for the current state-of-the-art. And in the underlying report [18] we made an effort at a systematic representation.

We introduce Green-Tao numbers \( \text{grt}_m(k) \) (“GT-numbers”; see Definition 3), which are defined as the \( \text{vdW} \)-numbers but using the first \( n \) prime numbers instead of the first \( n \) natural numbers. The existence of these numbers is given by the celebrated Green-Tao Theorem ([7]). We are concerned here also with the “mixed” GT-numbers \( \text{grt}_m(k_1, \ldots, k_m) \) (with \( \text{grt}_m(k) = \text{grt}_m(k, \ldots, k) \)). In Theorem 4 we show that transversal extension GT-numbers, which are of the form \( \text{grt}_{m+p}(2, \ldots, 2, k_1, \ldots, k_p) \), grow only linearly in \( m \). In the remainder of the article we are concerned with computing “all feasible” GT-numbers (computable within up to, say, a week by a single processor with the best SAT method).

For binary parameter tuples \((m = 2 \text{ above})\) the problem of computing \( \text{vdW} \)-or GT-numbers has a canonical translation to (boolean) SAT problems. For \( m > 2 \) we still have a canonical translation into non-boolean SAT problems, as is the case in general for hypergraph colouring problems (see [16]), but for using standard (boolean) SAT solvers the problem of a boolean translation arises. In Section 3 we introduce the (general) generic translation scheme, with seven natural instances, amongst them the well-known direct and logarithmic translations. As it turns out, in nearly all cases for all solver types the weak nested translation (introduced in [14]) performed far best, with the only exception that for relatively large numbers of colours the logarithmic translation was better.

For this (initial) phase of investigations into GT-numbers we only used “off-the-shelves” SAT solvers, establishing the “ground level” by providing the best solvers for the various parameter ranges. For over one year on average 10 processors were running, with a lot of manual interaction and adjustment to find the right solvers and translations, and to set the parameters (most basic the number of vertices), establishing the basic Green-Tao numbers. All generators and the details of the computations are available in the open-source research platform OKlibrary (see [13]). See Section 4 for the results of these computations. We conclude this article by a discussion of research directions in Section 5.

\[1\] http://www.ok-sat-library.org