Exploiting graph symmetry in the computation of chromatic number

Leonard Soicher

Queen Mary University of London

GAP Days Spring 2019, Martin Luther University of Halle-Wittenberg

The GRAPE package for GAP

- GRAPE is a GAP package for computing with graphs together with associated groups of automorphisms.
- It is designed for applications in algebraic graph theory, permutation group theory, design theory, and finite geometry.
- In GRAPE, a graph `gamma` comes together with a subgroup `gamma.group` of the automorphism group of `gamma`, and `gamma.group` is used to store the graph compactly and to speed up computations. This group is usually set by GRAPE when the graph is constructed, but may also be specified by the user.
- GRAPE also provides seamless interfaces to both *nauty* and *bliss* for computing automorphism groups of graphs and testing graph isomorphism.

GAP permutation group machinery is at the heart of GRAPE, and GRAPE should benefit from improvements in the GAP functionality for

- orbits (both for standard and user-defined actions);
- the image of a group G acting on the concatenation of specified G-orbits;
- point stabilizers;
- bases and strong generating sets;
- set stabilizers;
- canonical and minimal images of sets. (I currently use Steve Linton's `SmallestImageSet` function for this. Given $G \leq \mathrm{Sym}(n)$ and a subset A of $\{1, \ldots, n\}$, this function determines the lexicographically least G-image of A. I am also considering the functionality in the images package authored by Chris Jefferson, Markus Pfeiffer, Rebecca Waldecker and Eliza Jonauskyte.)

CompleteSubgraphsOfGivenSize

GRAPE has well-established powerful functionality for classifying cliques (the vertex-sets of complete subgraphs) in a graph.

For example, in GRAPE, where `gamma` is a (simple) GRAPE graph and `k` is a non-negative integer, the function call

```
CompleteSubgraphsOfGivenSize(gamma, k, 2, true)
```

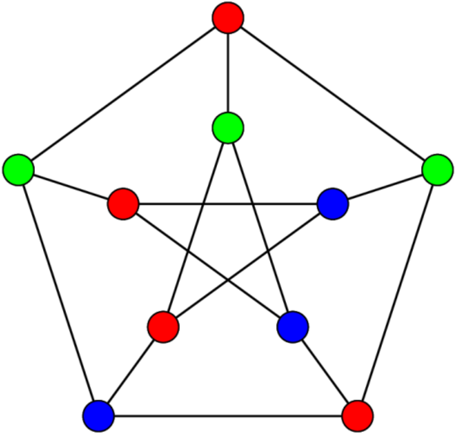
returns a set of `gamma.group` orbit-representatives of all the maximal cliques of size `k` in `gamma`.

Proper vertex-colourings

There is much recent functionality in GRAPE for proper vertex-colouring.

- A *proper vertex-colouring* of a graph is a labelling of the vertices of the graph by elements from a set of *colours*, such that any two vertices joined by an edge are labelled with different colours.
- Where k is a non-negative integer, a *vertex k-colouring* of a graph is a proper vertex-colouring using at most k colours.
- A *minimum vertex-colouring* of a graph Γ is a vertex k-colouring with k as small as possible, and the *chromatic number* $\chi(\Gamma)$ is the number of colours used in a minimum vertex-colouring of Γ.

A minimum vertex-colouring of the Petersen graph



https://graphtheoryinlatex.wordpress.com/2010/02/18/a-coloring-of-the-petersen-graph-2/

Minimum vertex-colourings in GRAPE

The method used in GRAPE for the determination of a minimum vertex-colouring of a graph Γ (and hence the chromatic number of Γ) is a binary search for the least k for which a vertex k-colouring of Γ exists, together with the determination of a vertex k-colouring for this least k.

The GRAPE function VertexColouring

In GRAPE $\geq$ 4.8, where `gamma` is a (simple) GRAPE graph and `k` is a non-negative integer, the function call

```
VertexColouring(gamma, k)
```

returns a vertex `k`-colouring of `gamma` (as a list of positive integers, indexed by the vertices of `gamma`, such that the *i*-th list element is the colour of vertex *i*) if such a colouring exists; otherwise, the special value `fail` is returned.

- Note that, up to the naming of the colours, the vertex k-colourings of Γ are in one-to-one correspondence with the partitions of the vertex set of the complement $\bar{\Gamma}$ of Γ into at most k cliques (the parts in such a partition are the colour classes of a vertex k-colouring of Γ).

- The approach in GRAPE to find a vertex k-colouring of Γ is to perform a backtrack search for the "least" (defined later) ordered partition $(C_1, C_2, \ldots, C_m)$ of the vertices of $\bar{\Gamma}$ into cliques, such that $m \leq k$.

- The search will either find such a partition or prove no such partition exists. [For the puposes of this talk, we will ignore some heuristics that are used which mean that, in practice, a vertex k-colouring may be found that does not correspond to the least partition of the vertices of $\bar{\Gamma}$ into at most k cliques.]

A well ordering of the subsets of $\{1, \ldots, n\}$

Let n be a non-negative integer, and let $A = \{a_1, \ldots, a_r\}$ and $B = \{b_1, \ldots, b_s\}$ be subsets of $\{1, \ldots, n\}$, with $a_1 < \cdots < a_r$ and $b_1 < \cdots < b_s$. We define

$$A \trianglelefteq B$$

to mean either $r > s$, or $r = s$ and $(a_1, \ldots, a_r) \leq (b_1, \ldots, b_r)$ in lexicographic order (w.r.t. the usual $\leq$ on the integers).

For example, $\{3, 5, 6\} \trianglelefteq \{2, 3\}$, but $\{3, 4, 7\} \trianglelefteq \{3, 5, 6\}$.

A well ordering of the sequences of length at most k of subsets of $\{1, \ldots, n\}$

Let k, n be a non-negative integers and let $\mathcal{A} = (A_1, \ldots, A_t)$ and $\mathcal{B} = (B_1, \ldots, B_u)$ be sequences of subsets of $\{1, \ldots, n\}$, with $t, u \leq k$. We define

$$\mathcal{A} \trianglelefteq \mathcal{B}$$

to mean that $(A_1, \ldots, A_t)$ is less than or equal to $(B_1, \ldots, B_u)$ in lexicographic order, with respect to the order $\trianglelefteq$ on subsets of $\{1, \ldots, n\}$.

For example, $(\{3, 4, 7\}, \{3, 5, 6\}, \{7, 8\}) \trianglelefteq (\{3, 4, 7\}, \{2, 3\})$.

The least vertex k-colouring

- Now let Γ be a graph with (non-empty) vertex set $V(\Gamma) = \{1, \ldots, n\}$, such that $V(\Gamma)$ can be partitioned into at most k cliques of Γ, for a given positive integer k.
- Then there is a unique least ordered such partition $(C_1, \ldots, C_m)$ with respect to $\trianglelefteq$.
- We now consider some properties of this least ordered partition, which give us very useful constraints on partial solutions in a backtrack search.

Constraints on partial solutions

Let $(C_1, \ldots, C_m)$ be the least ordered partition of $V(\Gamma)$ into cliques, with respect to $\trianglelefteq$, such that $m \leq k$. Let $\Gamma_1 := \Gamma$, let $G_1 := \mathrm{Aut}(\Gamma)$, and for $i = 2, \ldots, m$, let Γ_i be the subgraph of Γ_{i-1} induced on $V(\Gamma_{i-1}) \setminus C_{i-1}$ and let G_i be the (setwise) stabilizer in G_{i-1} of C_{i-1}. Then

- for $i = 1, \ldots, m$, C_i is a non-empty maximal clique of Γ_i, with $(k - i + 1)|C_i| \geq |V(\Gamma_i)|$, and if $(k - i + 1)|C_i| = |V(\Gamma_i)|$ then $m = k$ and $\{C_i, C_{i+1}, \ldots, C_m\}$ is a partition of $V(\Gamma_i)$ into maximal cliques of Γ_i, each of size $|C_i|$;

- for $i = 2, \ldots, m$, for $j = 1, \ldots, i - 1$, for each maximal clique C in Γ_j containing C_i, we have $|C| \leq |C_j|$ and if $|C| = |C_j|$ then the lexicographically least set in the G_j-orbit of C is not lexicographically less than C_j (in the search, currently only checked for $j = i - 1$);

- for $i = 1, \ldots, m$, C_i is the lexicographically least set in its G_i-orbit.

Final remarks

- I would like extensive functionality for canonical and minimal images (of sets) to be part of the standard GAP permutation group machinery.
- Included in this should be a function to determine, given $G \le \mathrm{Sym}(n)$ and subsets A, B of $\{1, \ldots, n\}$, whether (with respect to lexicographic order) the least G-image of A is less than B (without necessarily determining the least G-image of A).
- If an ordering other than lexicographic order is used for sets, the user should be able to access this ordering, say for ordering group orbits.