

A Separation Algorithm for the Clique Partitioning Problem

Adam N. Letchford¹ and Michael M. Sørensen²

¹ Department of Management Science, Lancaster University,
Lancaster LA1 4YX, UK. E-mail: A.N.Letchford@lancaster.ac.uk
² Department of Economics and Business Economics, Aarhus University, Fuglesangs
Allé 4, DK-8210 Aarhus V, Denmark. E-mail: mim@econ.au.dk.

1 Introduction

Given a complete graph $G = (V, E)$, with a rational weight x_e for each edge $e \in E$, the *clique partitioning problem* (CPP) calls for a partition of the vertex set V into cliques, such that the sum of the weights of the edges that have both end-vertices in the same clique is maximised [10, 11].

The standard 0-1 LP formulation of the CPP is as follows [7, 10]. For each edge $e = \{i, j\} \in E$, let x_e be a binary variable, taking the value 1 if and only if nodes i and j lie in the same set in the partition. We then have:

$$\max \quad \sum_{e \in E} w_e x_e \quad (1)$$

$$\text{s.t. } x_{ik} + x_{jk} - x_{ij} \leq 1 \quad (\{i, j\} \in E, k \in V \setminus \{i, j\}) \quad (2)$$

$$x_e \in \{0, 1\} \quad (e \in E). \quad (3)$$

The convex hull of solutions to (1)–(3) is called the *clique partitioning polytope*. The constraints (2) are called *transitivity* inequalities.

2 Known Inequalities and Separation Algorithms

Grötschel & Wakabayashi [8] showed that the transitivity inequalities (2) define facets, along with the non-negativity inequalities $x_e \geq 0$ for all e . They also discovered several other families of facet-defining inequalities, called *2-partition*, *2-chorded odd cycle* (2-COC), *2-chorded path* and *2-chorded even wheel* inequalities. Since then, several other families of valid and facet-defining inequalities have been discovered, such as:

- *odd wheel* (OW) and *odd bicycle wheel* (OBW) inequalities [4];
- *clique-web* inequalities, which include the 2-partition, OW and OBW inequalities as special cases [5];
- *odd closed walk* (OCW) inequalities, which include the 2-COC and OW inequalities as special cases [3, 12];
- *weighted* (s, T) inequalities [14];
- *q-chorded cycle* inequalities, which include the 2-COC inequalities as a special case [13];

- *odd gadget cycle* (OGC) inequalities, which include the OCW inequalities as a special case [9].

For a given family of inequalities to be used in a branch-and-cut algorithm, one must have a good (exact or heuristic) separation algorithm for that family. It is known that the separation problem can be solved in polynomial time for the transitivity inequalities [7]; OW and OBW inequalities [5]; OCW inequalities [3, 12] and OGC inequalities [9]. There are also some good separation heuristics for the 2-partition inequalities [7, 15] and weighted (s, T) inequalities [14].

For what follows, we recall that the q -chorded cycle inequalities take the form

$$\sum_{i=1}^c x(v_i, v_{i+1}) - \sum_{i=1}^c x(v_i, v_{i+q}) \leq c - \lceil c/q \rceil, \quad (4)$$

where (a) q and c are integers such that $q \geq 2$, $c > 2q$ and c is not a multiple of q , (b) $v_1, \dots, v_c$ are distinct vertices, and (c) indices are taken modulo c .

Andres *et al.* [1] call the q -chorded cycle inequalities with $c = 2q + 1$ *half-chorded odd cycle* (HCOC) inequalities. They showed that the HCOC inequalities define facets. They also claimed that the separation problem for the HCOC inequalities can be solved in polynomial time, by reduction to a minimum-weight odd cycle problem (MOC) in a suitable edge-weighted auxiliary graph. However, their algorithm contains an error, since the edge-weights in their auxiliary graph can be negative, whereas the MOC can only be solved in polynomial time if the edge-weights are non-negative [2, 6].

3 The New Separation Algorithm

We do not know whether there exists a polynomial-time separation algorithm for the HCOC inequalities, but we will now give a polynomial-time separation algorithm for a family of valid inequalities that *includes* all HCOC inequalities. First, we change the node labels in (4) to re-write the inequality as:

$$\sum_{i=1}^c x(v_i, v_{i+1}) - \sum_{i=1}^c x(v_i, v_{i+2}) \geq 3 - c. \quad (5)$$

It is not hard to see that (5) in turn is equivalent to:

$$\sum_{i=1}^c \left(x(v_i, v_{i+1}) + x(v_{i+1}, v_{i+2}) + 2 - 2x(v_i, v_{i+2}) \right) \geq 6. \quad (6)$$

We now construct an auxiliary *directed* graph, which we call D . We have one node in D for each ordered pair (i, j) of nodes in V . Then, for each ordered triple (i, j, k) of nodes in V , we have an arc in D going from node (i, j) to node (j, k) . The weight of the arc is set to $x_{ij}^* + x_{jk}^* + 2 - 2x_{ik}^*$, where $x^* \in [0, 1]^E$ is the point to be separated. Note that these arc weights are non-negative, as desired.

By construction, each HCOC inequality in the form (6) corresponds to an odd directed cycle (dicycle) in D , and such an inequality is violated by x^* if and only if that dicycle has a weight less than 6. Note however that not all odd dicycles in D correspond to HCOC inequalities. This is because a single node in V could potentially appear in the label of more than one node in the dicycle. Fortunately, this is not a problem, because we will show that the inequality (6) remains valid (though not necessarily facet-defining) even in this case.

We remark that D has $O(n^2)$ nodes and $O(n^3)$ arcs. Using the approach in [2, 6] to solve the minimum-weight odd dicycle problem, and a decent implementation of Dijkstra's shortest-path algorithm, we obtain an overall running time of $O(n^5)$. This is the same running time as for the existing separation algorithms for OBW, OCW and OGC inequalities [3, 5, 9, 12].

We hope to be able to present some preliminary computational results during the talk.

References

1. B. Andres, S.D. Gregorio, J. Irmay & J.-H. Lange (2023) A polyhedral study of lifted multicuts. *Discr. Optim.*, 47, article 100757.
2. F. Barahona & A.R. Mahjoub (1986) On the cut polytope. *Math. Program.*, 36, 157–173.
3. A. Caprara & M. Fischetti (1996) $\{0, \frac{1}{2}\}$ -Chvátal-Gomory cuts. *Math. Program.*, 74, 221–235.
4. S. Chopra & M.R. Rao (1993) The partition problem. *Math. Program.*, 59, 87–115.
5. M. Deza, M. Grötschel & M. Laurent (1992) Clique-web facets for multicut polytopes. *Math. Oper. Res.*, 17, 981–1000.
6. A.H.M. Gerards & A.J. Schrijver (1986) Matrices with the Edmonds–Johnson property. *Combinatorica*, 6, 365–379.
7. M. Grötschel & Y. Wakabayashi (1989) A cutting plane algorithm for a clustering problem. *Math. Program.*, 45, 59–96.
8. M. Grötschel & Y. Wakabayashi (1990) Facets of the clique partitioning polytope. *Math. Program.*, 47, 367–387.
9. A.N. Letchford & A.N. Vu (2021) Facets from gadgets. *Math. Program.*, 185, 297–314.
10. J.F. Marcotorchino (1981) *Aggregation of Similarities in Automatic Classification* (in French). Doctoral Thesis, Université Paris VI.
11. J.F. Marcotorchino & P. Michaud (1981) Heuristic approach to the similarity aggregation problem. *Meth. Oper. Res.*, 43, 395–404.
12. R. Müller (1996) On the partial order polytope of a digraph. *Math. Program.*, 73, 31–49.
13. R. Müller & A.S. Schulz (2002) Transitive packing: a unifying concept in combinatorial optimization. *SIAM J. Optim.*, 13, 335–367.
14. M. Oosten, J.H.G.C. Rutten & F.C.R. Spieksma (2001) The clique partitioning problem: facets and patching facets. *Networks*, 38, 209–226.
15. M.M. Sørensen (2020) A separation heuristic for 2-partition inequalities for the clique partitioning problem. *Working paper*, Department of Economics and Business Economics, Aarhus University.