
Two useful computational tricks for Quadratic Programming: hybrid SDP bounding procedures and a new linearisation technique

Fabio Furini^{*1} and Emiliano Traversi²

¹Laboratoire d'analyse et modélisation de systèmes pour l'aide à la décision (LAMSADE) – CNRS : UMR7024, Université Paris IX - Paris Dauphine – Place de Lattre de Tassigny 75775 PARIS CEDEX 16, France

²Laboratoire d'Informatique de Paris-Nord (LIPN) – Université Paris XIII - Paris Nord, CNRS : UMR7030, Institut Galilée – Institut Galilée 99, avenue J.B Clément 93430 VILLETANEUSE, France

Résumé

Quadratic programming problems have received an increasing amount of attention in the recent years, both from the theoretical and practical point of views. For this class of problems, we studied two different useful computational tricks.

The first one is the exploitation of Semidefinite Programming (SDP) relaxation within the framework provided by Mixed Integer Nonlinear Programming (MINLP) solvers. We included the SDP relaxation in a state-of-the-art MINLP solver as an additional bounding technique and demonstrated that this idea could be computationally useful. The Quadratic Stable Set Problem is adopted as the case study. The tests indicate that the Hybrid SDP Bounding Procedure allows an average 50% cut of the overall computing time and a cut of more than one order of magnitude for the branching nodes.

The second one is a new linearisation technique. We computationally prove that the new formulation, called Extended Linear Formulation, can be effective for different classes of problems in practice. Our tests are based on two sets of classical BQPs from the literature, i.e., the Unconstrained BQP and the Maximum Cut of edge-weighted graphs. Finally we discuss the relations between the Linear Programming relaxations of the different linearisation techniques presented and we discuss the elimination of constraint redundancy which is effective at speeding up the computational convergence.

Mots-Clés: Binary Quadratic Programming, Max Cut, Quadratic Stable Set Problem, Semidefinite Programming, Bounding Procedure

^{*}Intervenant