Discrete Optimization 22 (2016) 291–311

Contents lists available at ScienceDirect

Discrete Optimization

www.elsevier.com/locate/disopt

Integer programming as projection

H.P. Williams[a], J.N. Hooker[b,*]

[a] London School of Economics, United Kingdom
[b] Carnegie Mellon University, United States

A R T I C L E   I N F O

Article history:
Received 13 February 2015
Received in revised form 14 August 2016
Accepted 16 August 2016
Available online 10 September 2016

Keywords:
Integer programming
Projection
Duality
Value function

A B S T R A C T

We generalise polyhedral projection (Fourier–Motzkin elimination) to integer programming (IP) and derive from this an alternative perspective on IP that parallels the classical theory. We first observe that projection of an IP yields an IP augmented with linear congruence relations and finite-domain variables, which we term a *generalised IP*. The projection algorithm can be converted to a branch-and-bound algorithm for generalised IP in which the search tree has bounded depth (as opposed to conventional branching, in which there is no bound). It also leads to valid inequalities that are analogous to Chvátal–Gomory cuts but are derived from congruences rather than rounding, and whose rank is bounded by the number of variables. Finally, projection provides an alternative approach to IP duality. It yields a value function that consists of nested roundings as in the classical case, but in which ordinary rounding is replaced by rounding to the nearest multiple of an appropriate modulus, and the depth of nesting is again bounded by the number of variables. For large perturbations of the right-hand sides, the value function is shift periodic and can be interpreted economically as yielding "average" shadow prices.

1. Introduction

We propose an alternative perspective on integer programming that is based on projection. It begins with the observation that the projection of an integer programming (IP) problem is not an IP problem. More precisely, the projection of an IP problem's feasible set onto a subset of variables is not the feasible set of an IP. It is the feasible set of a disjunction of constraint sets, each consisting of an IP problem and congruence relations. The congruence relations define sublattices of the integer lattice. This suggests that an IP problem can be viewed more generally as defined over sublattices of the integer lattice, rather than over the entire integer lattice as in conventional IP. We will call this a *generalised IP problem*, whose projection onto any subset of variables is another generalised IP problem.

* Corresponding author.
E-mail addresses: H.P.Williams@lse.ac.uk (H.P. Williams), jh38@andrew.cmu.edu (J.N. Hooker).

http://dx.doi.org/10.1016/j.disopt.2016.08.004

The projection problem for generalised IPs can be solved by introducing integer auxiliary variables with finite domains, and taking advantage of a generalised Chinese Remainder theorem. The function of the auxiliary variables is to define the sublattices. Projecting out all the original variables transforms the optimisation problem to one that minimises over a system of congruence relations that involve only the auxiliary variables. A problem of optimising over possibly infinite domains is therefore transformed to one of optimising over finite domains.

This perspective leads to an alternative theory of cutting planes, branching algorithms, and IP duality. We introduce "congruence cuts", which are analogous to Chvátal–Gomory cuts, except that they are derived from a linear combination strengthened by a congruence relation, rather than a linear combination strengthened by rounding. We use the projection algorithm to show that their rank is bounded by the number of variables. This contrasts with the classical Chvátal rank, which has no bound related only to the number of variables [1].

In addition, we show that the projection algorithm can be converted to a branching algorithm that branches on integer auxiliary variables rather than the original integer variables, and in which the possible branches are defined by congruence relations. The depth of the tree is again bounded by the number of variables, whereas a conventional branching tree has unbounded depth.

Finally, by applying the projection algorithm to an IP problem with general right-hand sides, we obtain a value function that is analogous to a Gomory function [2,1] in that it contains nested rounding operations. However, rather than rounding to the nearest integer, one rounds to the nearest multiple of an appropriate modulus. In contrast to a value function obtained by Gomory's method, the depth of nesting (which is analogous to cutting plane rank) is bounded by the number of variables, and the function can be obtained by one pass through the model. We show that this value function is shift-periodic for large perturbations of the right-hand sides, which leads to an economic interpretation and "average" shadow prices.

Our analysis is intended for problems with general integer variables, where number theoretic issues come into play. It is of particular interest when the variables have no upper bounds, because we show that the search trees, cutting plane proofs, and dual functions have bounded depth even when the variables are unbounded.

The idea of extending Fourier–Motzkin elimination to project the feasible set of an IP was originally proposed by one of us (Williams) in [3]. In that work, the elimination of each variable created a disjunction of linear systems and congruences, corresponding to the sublattices defined here. In the present paper, we introduce auxiliary variables to define the sublattices, as well as formally stating the projection algorithm and proving its correctness. This allows us to go beyond [3] and other previous work by developing an alternative theory of IP that includes cutting planes of bounded rank, a branching algorithm with bounded depth, and a duality theory in which the value functions have bounded nesting depth.

We begin with brief discussion of related work. We then review projection and duality in linear programming (LP), to clarify how it is generalised for the IP case. At this point we proceed to establish the results just described.

2. Related work

A result due to Meyer [4] implies that branching has bounded depth for IP if the branch points are properly chosen. Meyer showed that one can construct a finite box within which an optimal solution lies (if one exists), and this allows one to find the optimum by branching on integer values that lie within the confines of the box. Our bound is based on a quite different argument and does not require the construction of a box. Lenstra's method [5] for solving an IP problem with a fixed number of variables in polynomial time also leads to a depth bound equal to the number of variables, but it relies on ellipsoidal approximation and lattice basis reduction rather than projection.

The concept of the value function of a mathematical programme is due to Blair and Jeroslow [2]. Here we show that the value function can be expressed in a different form with bounded nesting depth. A forerunner