Available online at www.sciencedirect.com

ScienceDirect

Fuzzy Sets and Systems 272 (2015) 30–46

FUZZY sets and systems

www.elsevier.com/locate/fss

Short rational generating functions for solving some families of fuzzy integer programming problems

Víctor Blanco [a,*], Justo Puerto [b]

[a] Dep. of Quantitative Methods for Economics & Business, Universidad de Granada, Spain
[b] Instituto Universitario de Investigación Matemática (IMUS), Universidad de Sevilla, Spain

Received 4 March 2010; received in revised form 21 April 2014; accepted 4 November 2014

Available online 8 November 2014

Abstract

In this paper we present new complexity results for solving single and multiobjective fuzzy integer programs with soft constraints and fuzzy objective function coefficients. Our method is based on the use of Barvinok's short rational generating functions of convenient transformations of the fuzzy problems to crisp ones. This analysis allows us to provide new insights on the structure of these families of fuzzy programs.
Published by Elsevier B.V.

Keywords: Fuzzy mathematical programming; Integer programming; Theoretical complexity; Generating functions

1. Introduction

Integer linear programming (IP) deals with the problem of minimizing or maximizing a linear function over the solutions of a diophantine system of inequalities. IP is a very important area of activity because it is a natural tool to model and handle many real life situations, as for instance management and efficient use of resources: distribution of goods, production scheduling, machine sequencing, capital budgeting, etc. Furthermore, from a mathematical point of view many interesting combinatorial and geometrical problems in Graph Theory and Logic can be seen as integer programs. For these reasons, integer programming has attracted many researchers from different areas and there is nowadays an extensive body of literature (textbooks and publications) exclusively devoted to the analysis and resolution of these problems, both from a theoretical and a practical viewpoint. The interested reader is referred to the classical textbooks by Schrijver [1], Nemhauser and Wolsey [2] or the more recent by Sierksma [3] among many others, for further details. Different methodologies have been designed to solve integer programming problems, among them, we mention the widely used algorithms based on branch-and-bound, cutting planes, branch-and-cut methods or

* Corresponding author at: Dep. of Quantitative Methods for Economics & Business, Facultad de Ciencias Económicas y Empresariales, Campus Cartuja, 18011 Granada, Spain. Tel.: +34 958249637.
E-mail addresses: vblanco@ugr.es (V. Blanco), puerto@us.es (J. Puerto).

http://dx.doi.org/10.1016/j.fss.2014.11.003
0165-0114/Published by Elsevier B.V.

dynamic programming. However, it is well-known that solving an IP is generally NP-hard when the dimension is part of the input, although polynomial-time solvable in fixed dimension (see [1,4]).

Decision-makers usually find vague information when trying to model a real-world situation as a mathematical programming problem. This imprecision, which causes difficulties in modeling, can be considered by assuming that some of the elements in the formulation are fuzzy entities (see [5,46]). In [6, Chapter 4] the authors present a detailed list of options to "fuzzify" a mathematical programming problem. When some of the elements in an IP are fuzzy, the problem falls into a different class that it is usually called Fuzzy Integer Programming (FIP). This latter class of problems has been widely studied in the literature (see for instance, [7–15] among many others). However, its theoretical complexity has not been yet stated. It is clear that solving a FIP is, in general, much harder than solving an IP, since a standard integer program is a very particular case of FIP. In particular, FIP is also NP-hard. Nevertheless, although it is well-known that IP can be solved in polynomial-time when one fixes the dimension of the space of variables, there are not known results concerning the complexity of FIP when the dimension is fixed.

In this paper we provide new exact algorithms for solving some important models of FIP problems (namely those described in [12,16] and [17]) and we analyze their theoretical complexity. Both the results and the tools are new in this field. Furthermore, although some of the proofs presented in this paper are based on previous results by Barvinok [18], they are not trivial extensions since Barvinok's theory states the polynomial-time complexity of (crisp) IP, but not of its fuzzy versions. For each of the models that we consider, we need to perform convenient transformations that allow us to apply short rational generating functions results (not only Barvinok's results [18] but also some others related to the complexity of multiobjective integer programming [19]). In particular, we deal with three different fuzzifications of integer programming problems. One is based on using fuzzy sets to represent the vagueness of the inequality relations in the constraints of the integer program, usually called *flexible programming* (see [20,21] and the references therein); the second one, on the consideration of fuzzy coefficients in the linear objective function (see Section 4.4 in [22] or [23,24]), i.e., when the imprecision concerns the costs associated to the variables in the model; and finally, single and multiobjective IP problems where both the constraints and the coefficients of the objective function are fuzzy elements. We give polynomiality results for FIP similar to those proved by Lenstra [25] about the polynomiality of (crisp) IP in fixed dimension. In our approach, we perform some transformations of the considered fuzzy problems to multiobjective integer programs. These transformations allow us to apply new tools borrowed from the theory of short rational generating functions to prove the polynomial complexity of several models of fuzzy integer programming. The use of short rational generating functions for solving fuzzy optimization problems is instrumental and new; and it is the milestone that allows us to prove the new complexity results.

The paper is organized as follows. Section 2 is devoted to recall some previous notions and results about short rational generating functions (SRGF) of rational polytopes, in particular, the main results concerning the complexity of computing them and its application for solving multiobjective integer programming problems. We present, in Section 3, complexity results for fuzzy integer programs where the fuzziness is induced by considering soft constraints. In Section 4 we analyze integer programs with fuzzy coefficients in the objective functions. At the end of that section we state similar complexity results for single and multiobjective fuzzy integer programs where both the coefficients of the objective functions and the constraints are fuzzy. Finally, in Section 5 we draw some conclusions about the results presented in this paper.

2. Preliminaries

2.1. Short rational generating functions

Short rational generating functions (SRGF) were first used by Barvinok [18] for counting integer points inside rational bounded polyhedra (polytopes), based on the previous geometrical paper by Brion [26]. The main idea is encoding those integral points in a rational function with as many variables as the dimension of the space where the body lives. Let $P \subset \mathbb{R}^d_+$ be a given convex polyhedron, the integral points may be expressed in a formal sum $f(P, z) = \sum_\alpha z^\alpha$ with $\alpha = (\alpha_1, \ldots, \alpha_d) \in P \cap \mathbb{Z}^d$, where $z^\alpha = z_1^{\alpha_1} \cdots z_d^{\alpha_d}$. Barvinok's goal was representing that formal sum of monomials in the multivariate polynomial ring $\mathbb{R}[z_1, \ldots, z_d]$, as a "short" sum of rational functions in the same variables. Actually, Barvinok presented a polynomial-time algorithm when the dimension, n, is fixed, to compute those functions. A clear example to illustrate that approach is the polytope $P = [0, N] \subset \mathbb{R}$: the long