European Journal of Operational Research 000 (2015) 1–15

Contents lists available at ScienceDirect

European Journal of Operational Research

journal homepage: www.elsevier.com/locate/ejor

Decision Support

Finding the nucleoli of large cooperative games

Tri-Dung Nguyen [a,*], Lyn Thomas [b]

[a] Mathematical Sciences and Business School, University of Southampton, Southampton SO17 1BJ, United Kingdom
[b] School of Business, University of Southampton, Southampton SO17 1BJ, United Kingdom

ARTICLE INFO

Article history:
Received 30 June 2014
Accepted 12 August 2015
Available online xxx

Keywords:
Nucleolus
Cooperative game
Multi-level programming
Payoff distribution
Constraint generation
Lexicographical minimization

ABSTRACT

The nucleolus is one of the most important solution concepts in cooperative game theory as a result of its attractive properties - it always exists (if the imputation is non-empty), is unique, and is always in the core (if the core is non-empty). However, computing the nucleolus is very challenging because it involves the lexicographical minimization of an exponentially large number of excess values. We present a method for computing the nucleoli of large games, including some structured games with more than 50 players, using nested linear programs (LP). Although different variations of the nested LP formulation have been documented in the literature, they have not been used for large games because of the large size and number of LPs involved. In addition, subtle issues such as how to deal with multiple optimal solutions and with tight constraint sets need to be resolved in each LP in order to formulate and solve the subsequent ones. Unfortunately, this technical issue has been largely overlooked in the literature. We treat these issues rigorously and provide a new nested LP formulation that is smaller in terms of the number of large LPs and their sizes. We provide numerical tests for several games, including the general flow games, the coalitional skill games and the weighted voting games, with up to 100 players.

1. Introduction

The nucleolus is one of the most important solution concepts for cooperative games with transferable utilities. It represents a way to distribute the reward (or cost) among the players involved in a way that lexicographically minimizes the excess values (i.e. dissatisfaction levels) of all coalitions. The nucleolus was introduced in 1969 by Schmeidler (1969) as a solution concept with attractive properties - it always exists (if the imputation is non-empty), it is unique, and it lies in the core (if the core is non-empty). We review concepts in cooperative game theory and their mathematical definitions in Section 2.1. The nucleolus concept has been used in many different applications. For example, in the banking industry, groups of banks enter into an agreement for their customers to use ATM machines owned by any bank in the same group. The nucleolus is then used to suggest how the cost of installing and maintaining those ATM machines can be shared among the banks (see Gow & Thomas (1998)). It has also been applied to insurance premium setting (Lemaire (1991)) and to network cost sharing (Deng, Fang, and Sun (2009) Granot and Huberman (1984) Granot and Maschler (1998)), among many other applications.

Despite the desirable properties that the nucleolus has, its computation is, however, very challenging because the process involves the lexicographical minimization of 2^n excess values, where n is the number of players. Kohlberg (1971) provides a necessary and sufficient condition for an imputation to be the nucleolus. However, the criterion requires forming the coalition array of all 2^n possible coalitions and then making sure 2^n linear inequalities are satisfied. The analytical form of the nucleolus is only available for games with three players (see Leng & Parlar (2010)). There are a small number of games whose nucleoli can be computed in polynomial time. These include the connected games in Solymosi and Raghavan (1994), the neighbor games in Hamers, Klijn, Solymosi, Tijs, and Vermeulen (2003), the cyclic permutation games in Solymosi, Raghavan, and Tijs (2005), and the flow games with unit capacities in Deng et al. (2009) Kern and Paulusma (2009) Potters, Reijnierse, and Biswas (2006). It has been shown that finding the nucleolus is NP-hard for many classes of games such as the utility games with non-unit capacities (Deng et al. (2009)) and the weighted voting games (Elkind, Goldberg, Goldberg, & Wooldridge (2007)). In fact, finding the core and the least core is NP-hard in supermodular games Schulz and Uhan (2010) and inventory centralization games Chen and Zhang (2009).

Kopelowitz (1967) suggests using nested linear programming (LP) to compute the kernel of a game. This encouraged a number of researchers to compute the nucleolus using linear programming. For example, Kohlberg (1972) presents a single LP with $O(2^n!)$

* Corresponding author. Tel.: +44 (0)23 8059 7759; fax: +44 (0)23 8059 5147.
E-mail addresses: T.D.Nguyen@soton.ac.uk, ngtridung@gmail.com (T.-D. Nguyen), L.Thomas@soton.ac.uk (L. Thomas).

http://dx.doi.org/10.1016/j.ejor.2015.08.017
0377-2217/

constraints. The number of constraints in the LP formulation of Owen (1974) is reduced to $O(4^n)$ but the coefficients get larger. The nucleolus can also be found by solving a sequence of LPs. However, the number of LPs involved is exponentially large (i.e. $O(4^n)$ in Maschler, Peleg, and Shapley (1979) and $O(2^n)$ in Sankaran (1991)). Potters, Reijnierse, and Ansing (1996) present another formulation that involves solving $(n-1)$ linear programs with, at most, (2^n+n-1) rows and (2^n-1) columns. The authors also develop a prolonged Simplex method for solving these large LPs and conduct numerical experiments for games with 10 players. Derks and Kuipers (1997) improve the implementation of the prolonged Simplex method in Potters et al. (1996) and provide numerical results for games with 20 players. Göthe-Lundgren, Jörnsten, and Värbrand (1996) attempted to apply a constraint generation framework to find the nucleoli of basic vehicle routing games. However, some of results are incorrect as has been pointed out by Chardaire (2001). The issue of having multiple optimal solutions in each LP was not considered in Göthe-Lundgren et al. (1996), but we are able to deal with that in Section 3.3. Fromen (1997) uses Gaussian elimination to improve the formulation of Sankaran (1991) and to reduce the number of LPs in the implementation. Nevertheless, the LPs are still extremely large when the number of players gets large and existing methods become intractable when n exceeds 20. In the nested LPs formulation, subsequent LPs are formed based on the optimal solutions of the previous LPs and the tight inequalities. One needs to be very careful if there are multiple optimal solutions to these LPs and if there are multiple coalitions with the same worst excess values. The paper provides a rigorous treatment of the nested LPs formulation which deals with these issues.

For each payoff distribution, there are 2^n excess values that correspond to 2^n possible coalitions. The nucleolus is the payoff distribution that lexicographically minimizes its excess values. Thus finding the nucleolus effectively is only possible if one can find the worst coalition(s) for a given imputation efficiently. Faigle, Kern, and Kuipers (2001) show that the nucleolus can be found in polynomial time if finding the worst coalition for a given imputation, i.e. the separation problem, can be done in polynomial time. Their method is based on the ellipsoid algorithm which, although theoretically has a polynomial runtime, does not perform well in practice. Our paper bridges this gap and presents a practical numerical procedure for computing the nucleolus. We test our algorithm with the coalitional skill games and the weighted voting game.

The key contributions of our work include:

- We present a nested LPs formulation for computing the nucleoli of cooperative games. Although the idea of using a nested LPs framework has been around for more than 40 years ago (Kopelowitz (1967)) with various reformulations having been proposed, these methods face several issues that will be described in detail in Section 2.4. The most critical issue among these is how to *handle multiple optimal solutions in each of the large LPs*. Dealing with multiple optimal solutions is often needed in multi-level programming and is often very challenging. We provide a concrete method for dealing with these issues in Sections 3.3 and 4.3.
- The size of our nested LPs formulation is smaller than other nested LPs formulation described in the literature as can be seen in Table 1. The number of LPs to be solved in our method is smaller than that in Maschler et al. (1979) and Sankaran (1991) while the number of columns in each LP is smaller than that in Potters et al. (1996). These features are results of our special way of handling tight coalitions and finding the minimal tight sets (the key idea in Theorem 3 and the main results in Theorem 1).
- We provide numerical computation for large and structured games with up to 100 players in the weighted voting games Aziz, Paterson, and Leech (2007) Chalkiadakis, Elkind, and Wooldridge (2011) and up to 75 players in the coalitional skill games Bachrach and Rosenschein (2008). This is a significant improvement compared to the literature where numerical results are shown for computing the nucleoli of games with at most 20 players.

Table 1
Comparison between our method and the literature.

Algorithms	*# LPs and their sizes*
Kohlberg (1972) in 1972	One LP with $O(2^n!)$ constraints
Owen (1974) in 1974	One LP with $O(4^n)$ constraints but the coefficients get large
Maschler et al. (1979) in 1979	$O(4^n)$ LPs, each with $O(2^n)$ rows and $n+1$ columns
Sankaran (1991) in 1991	$O(2^n)$ LPs, each with $O(2^n)$ rows and $n+1$ columns
Potters et al. (1996) in 1996	$n-1$ LPs, each with (2^n+n-1) rows and (2^n-1) columns
Our method	$n-1$ LPs, each with (2^n+n-1) rows and $(n+1)$ columns

In addition to these key contributions, we also apply a constraint generation framework for solving the large LPs. This gives hope to solving very large LPs as we don't have to rely on the simplex method to solve large scale LPs when $n \geq 25$. The constraint generation algorithm is not new and has been applied successfully in many areas including finding the solutions of cooperative games (e.g. Caprara & Letchford (2010)). Applying it to solving nested LPs though creates some challenging problems in keeping track of multiple discrete and continuous optimal solutions so that subsequent LPs can be formulated. Our approach is appropriate for large games and for combinatorial games where it is costly to calculate the characteristic values for all the possible coalitions. For example, in the flow games proposed by Kalai and Zemel (1982), the value of a coalition is the maximum flow that can be sent through the subnetwork using edges in the coalition only. In this case, it is time consuming to calculate all the 2^n possible values. Instead, we can incorporate the maximum flow problem into the constraint generation problem that is then solved only if needed. We also demonstrate how this can be done in other games such as the voting games and the coalitional skills games.

The structure of this paper is as follows: Section 2.1 provides the list of notations used throughout the paper and a review of important solution concepts in cooperative game theory such as the core, the least core, and the nucleolus. We present the idea behind the nested LPs and their formulation in Sections 2.2 and 2.3 with an illustrative example. The focus of this paper starts from Section 2.4 where the subtle issues relating to the nested LPs formulation are discussed in detail. The main contribution of this paper lies in addressing these issues in Section 3. We present the framework for finding the nucleolus in Section 3.2 and the idea of finding optimal solutions with minimal tight sets in Section 3.3. Our algorithm requires solving at most $(n-1)$ large LPs, each having $(n+1)$ columns and (2^n+n-1) rows as shown in Section 3.2. For large games, we present methods for solving the exponentially large LPs and for handling large sets of tight constraints. This includes a constraint generation method for solving these large LPs in Section 4.1 and the idea of using representative sets for keeping track of tight constraints in Section 4.3. Various numerical experiments are presented in Section 6 and conclusion is drawn in Section 7.

2. Finding the nucleolus using nested LPs: background and issues

2.1. Review of solution concepts in cooperative game theory and notations

Let n be the number of players and let $\mathcal{N} = \{1, 2, \dots, n\}$ be the set of all the players. A *coalition* $\mathcal{S}$ is a subset of the players, i.e. $\mathcal{S} \in \mathcal{C}$. Let $\mathcal{C} \equiv 2^{\mathcal{N}}$ be the set of all the possible coalitions. The *characteristic*