Mathematical Social Sciences 40 (2000) 227–235

mathematical social sciences

www.elsevier.nl/locate/econbase

Easy weighted majority games

Nilotpal Chakravarty[a], Anand Mohan Goel[b,1], Trilochan Sastry[c,*]

[a]*Indian Institute of Management, P.O. Box No 16757, P.O. Alipore, Calcutta 700 027, India*
[b]*E/139-A First Floor, 31st Cross Street, California Software Company Ltd., Besant Nagar, Madras 600 090, India*
[c]*Indian Institute of Management, Vastrapur, Ahmedabad 380 015, India*

Received November 1998; received in revised form September 1999; accepted November 1999

Abstract

In a weighted majority game each player has a positive integer weight and there is a positive integer quota. A coalition of players is winning (losing) if the sum of the weights of its members exceeds (does not exceed) the quota. A player is pivotal for a coalition if her omission changes it from a winning to a losing one. Most game theoretic measures of the power of a player involve the computation of the number of coalitions for which that player is pivotal. Prasad and Kelly [Prasad, K., Kelly, J.S., 1990. NP-completeness of some problems concerning voting games. International Journal of Game Theory 19, 1–9] prove that the problem of determining whether or not there exists a coalition for which a given player is pivotal is NP-complete. They also prove that counting the number of coalitions for which a given player is pivotal is #P-complete. In the present paper we exhibit classes of weighted majority games for which these problems are easy.

1. Introduction

In a weighted majority game with n players, indexed 1 to n, player i, $1 \leq i \leq n$, has a positive integer *weight* w_i and there is a positive integer *quota* q. Let $N = \{1, 2, \ldots, n\}$ denote the set of all players. Any subset of N is called a *coalition*. A coalition S of players is *winning* (*losing*) if $\Sigma_{j \in S}\, w_j > q$ ($\leq q$). Player i is *pivotal* for coalition S if $S \backslash \{i\}$ is losing whereas S is winning. Most game theoretic measures of the power of a given player, such as the Shapley Value and the Banzhaf–Coleman Index, involve the

*Corresponding author. Tel.: +91-79-640-7241; fax: +91-79-642-7896.
E-mail address: sastry@iimahd.ernet.in (T. Sastry)
[1]Currently at the University of Michigan Business School, Ann Arbor, MI 48109, USA.

0165-4896/00/$ – see front matter
PII: S0165-4896(99)00050-5

computation of the number of coalitions for which that player is pivotal. [Refer, for example, to Roberts (1976) or Owen (1995).]

Suppose that we are given a weighted majority game. Let us denote by $P_1(i)$ the problem of determining whether or not player i is pivotal for some coalition. The problem is equivalent to the following subset sum problem:

$$\text{Maximize} \sum_{j \in N \setminus \{i\}} w_j x_j \tag{1}$$

$$\text{Subject to} \sum_{j \in N \setminus \{i\}} w_j x_j \leq q, x_j \in \{0,1\} \, \forall j \tag{2}$$

Player i is pivotal for *at least one* coalition if and only if the maximum value of (1) subject to (2) is strictly greater than $q - w_i$. It was proved in Prasad and Kelly (1990) that $P_1(i)$ is NP-complete. [For definitions of NP-completeness and all other terminology related to complexity theory used in this paper, refer to Papadimitriou (1994).] This makes it very unlikely that there will be an efficient solution algorithm for $P_1(i)$, where by an efficient algorithm we mean one which solves the problem in time bounded by a polynomial in n.

We will denote by P_1 the related problem:

$$\text{Maximize} \sum_{j \in N} w_j x_j, \quad \text{subject to} \sum_{j \in N} w_j x_j \leq q, x_j \in \{0,1\} \, \forall j \tag{P_1}$$

We will denote by $P_2(i)$ the problem of determining the number of feasible solutions to (2). Observe that the number of coalitions for which player i is pivotal may be determined efficiently (i.e. in time polynomial in n) if we have available an algorithm which solves $P_2(i)$ in polynomial time for all q. We need merely run the algorithm twice, once to determine the number of feasible solutions to (2) and once more to determine the number of feasible solutions to:

$$\sum_{j \in N \setminus \{i\}} w_j \, x_j \leq q - w_i, x_j \in \{0,1\} \, \forall j$$

and subtract the latter from the former. However, Prasad and Kelly (1990) show that determining the number of coalitions for which player i is pivotal is a #P-complete problem. This makes the existence of any such polynomial time algorithm most unlikely. We will denote by P_2 the related problem of determining the number of feasible solutions to:

$$\sum_{j \in N} w_j x_j \leq q, x_j \in \{0,1\} \, \forall j$$

In the present paper we show that for certain choices of weights the problems $P_1(i)$ and $P_2(i)$ may be solved efficiently. For such weighted majority games it is therefore easy to determine whether or not a player is pivotal for some coalition and to count the number of coalitions for which she is pivotal. We assume that two weights can be added in constant time, and that we can ignore weights with many digits, and hence, the size of the input is n. For the remainder of the paper we assume that $w_1 \geq w_2 \geq \ldots \geq w_n$.

Sorting the weights to ensure this requires $O(n \log n)$ time. [Refer, for example, to Aho et al. (1974).]

2. Unbalanced weights

We will say that the set of weights $W = \{w_1, w_2, \ldots, w_n\}$ is *unbalanced* if $w_j \geq w_{j+1} + w_{j+2} + \ldots + w_n$, for all j, $1 \leq j < n$. Thus, for example, {42, 20, 10, 5, 3, 1} is unbalanced. Unbalanced weights occur in a completely different context in cryptography (refer, for example, to Konheim, 1981), though they are not given this name. It is evident that a subset of an unbalanced set is itself unbalanced.

Consider now the following *greedy algorithm* which outputs the *greedy solution* $g \in \{0,1\}^n$ to P_1.

Greedy Algorithm:

- *Initialization*: Set SUM $= 0$, $j = 1$. Go to Step 1.
- *Step 1*: If SUM $+ w_j \leq q$, then set $g_j = 1$, SUM $=$ SUM $+ w_j$. Else set $g_j = 0$. Set $j = j + 1$. Go to Step 2.
- *Step 2*: If $j = n + 1$, then output the vector g and SUM and stop. Else go to Step 1.

Problem P_1 is solved by the above greedy algorithm when the set of weights is unbalanced. This fact is apparently well known to cryptographers, and Konheim (1981, p. 304) provides a sort of informal proof of this by the way of an example. We provide below a quick formal proof.

First we prove a lemma which we will use several times in the paper. Note that the lemma is valid for all sets of weights W, not just unbalanced ones.

Lemma 1. *Let x be any feasible solution to P_1, different from the greedy solution g, such that $g_j = x_j$, for $j < i$, and $g_i \neq x_i$, i.e. let i be the least* (*i.e. left-most*) *index in which g differs from x. Then $g_i = 1$ and $x_i = 0$.*

Proof. $x_i = 1$ would imply that $\Sigma_{j<i} w_j g_j + w_i = \Sigma_{j<i} w_j x_j + w_i \leq q$. However, then the greedy algorithm must set $g_i = 1$ too, which contradicts the definition of i. Hence, $x_i = 0$, $g_i = 1$. □

Theorem 2. *The greedy algorithm solves Problem P_1 if the set of weights $\{w_1, w_2, \ldots, w_n\}$ is unbalanced.*

Proof. Consider any feasible solution x to P_1, different from the greedy solution g. Let i be the least (i.e. left-most) index in which g differs from x. Then $g_j = x_j$, for $j < i$, and $g_i \neq x_i$. By Lemma 1 $x_i = 0$, $g_i = 1$. Now:

$$\sum_{j \in N} w_j g_j \geq \sum_{j<i} w_j g_j + w_i \geq \sum_{j<i} w_j x_j + \sum_{j>i} w_i$$

since the set of weights is unbalanced, and hence:

$$\sum_{j\in N} w_j g_j \geq \sum_{j\in N} w_j x_j$$

since $x_i = 0$. This establishes the optimality of g. $\square$

The greedy algorithm clearly requires $O(n)$ elementary arithmetic operations. Since any subset of an unbalanced set is itself unbalanced we can solve $P_1(i)$ for *any given* i in $O(n)$ time by running the greedy algorithm on the set of weights $W\backslash\{w_i\}$. Consequently we can solve $P_1(i)$ for *all* i, in $O(n^2)$ time, by running the greedy algorithm n times. We can, however, do better.

Theorem 3. *$P_1(i)$ can be solved for every i in overall $O(n)$ time, if the set of weights $\{w_1, w_2, \ldots, w_n\}$ is unbalanced.*

Proof. Consider an execution of the greedy algorithm on the set of weights $W\backslash\{w_i\}$ and let y denote the vector output at the end. The first $i-1$ executions of Step 1 are identical to those of the same algorithm when run on the set of weights W. We must therefore have $y_j = g_j$ for all $j < i$. If $g_i = 0$ then the last $n-i$ executions of Step 1 must be identical too. In this case y is simply obtained from g by suppressing g_i and the optimal objective values of P_1 and $P_1(i)$ are identical.

Suppose next that $g_i = 1$. In this case, when Step 1 executes for the ith time, we have $\text{SUM} + w_{i+1} + w_{i+2} + \ldots + w_n \leq \text{SUM} + w_i \leq q$, and therefore the algorithm sets $y_j = 1$, $\forall j \geq i$. It follows that in this case optimal solution to $P_1(i)$ is $y = (y_1, y_2, \ldots, y_{i-1}, y_{i+1}, \ldots, y_n)$, where $y_j = g_j$, $\forall j < i$, and $y_j = 1$, $\forall j > i$. The optimal objective value of $P_1(i)$ in this case is $\Sigma_{j<i} w_j g_j + \Sigma_{j>i} w_i$. Now it is possible to compute all the partial sums $\Sigma_{j<i} w_j g_j$, $1 < i \leq n$ in $O(n)$ time. In fact, these sums are computed in the course of the greedy algorithm. It is only necessary to slightly modify the algorithm to keep track of these. The partial sums $\Sigma_{j>i} w_j$, $1 \leq i < n$ can also all be computed in time $O(n)$. It is clear from the above discussion that if these partial sums are all available the optimal objective value of each $P_1(i)$ can be determined using a *constant* number of arithmetic operations. The result now follows. $\square$

We next show that if W is unbalanced and $g = (g_1, g_2, \ldots, g_n)$, denotes the greedy solution to P_1, then the number of feasible solutions to P_1 is $\Sigma_1^n g_j 2^{n-j} + 1$. In other words it is the binary number $g_1 g_2 \ldots g_n + 1$. In the following proof and thereafter we use $|X|$ to denote the cardinality of any set X.

Theorem 4. *If the set of weights $\{w_1, w_2, \ldots, w_n\}$ is unbalanced and $g = (g_1, g_2, \ldots, g_n)$, denotes the greedy solution to P_1, then the number of feasible solutions to P_1 is $\Sigma_1^n g_j 2^{n-j} + 1$.*

Proof. Let S denote the set of feasible solutions to P_1. Then S is the disjoint union of the sets S_i, $1 \leq i \leq n$, where $S_i = \{x | x$ is a feasible solution to P_1, $g_j = x_j$, for $j < i$, and $g_i \neq x_i\}$, and the singleton $\{g\}$ and $|S|$ is clearly equal to $\Sigma_1^n |S_i| + 1$.

Now by Lemma 1 if $g_i = 0$ then $|S_i| = 0$. Suppose now that $g_i = 1$. Then each of the first $i-1$ coordinates of any solution in S_i must coincide with the corresponding coordinate of g and the ith coordinate must be zero. Since $g_i = 1$ $\Sigma_{j>i}$ $w_j \leq w_i \leq q - \Sigma_{j<i}$ w_j and so any of the 2^{n-i} possible assignments of 0's and 1's to the last $n-j$ coordinates yields a feasible solution to P_1. Hence in this case $|S_i| = 2^{n-i}$. The result follows. □

It follows trivially from Theorem 4 that P_2 can be solved in $O(n)$ time if the set of weights is unbalanced. Since any subset of an unbalanced set is also unbalanced $P_2(i)$ can also be solved in $O(n)$ time for any given i.

We conclude this section by remarking that a mild generalization of the above results is possible. Let g be the greedy solution to P_1. Let us call a set of weights $W = \{w_1, w_2, \ldots, w_n\}$ *generalized unbalanced* if $w_i \geq \Sigma_{j>i}$ w_j $(1 - g_j)$ $\forall i$, $1 \leq i < n$. (Equivalently the set of weights $W = \{w_1, w_2, \ldots, w_n\}$ is generalized unbalanced if $w_i \geq \Sigma \{w_j \mid j > i, g_j = 0\}$ $\forall i$, $1 \leq i < n$.) Each of Theorems 2, 3 and 4 remains valid if we replace the word 'unbalanced' by 'generalized unbalanced' in its statement. Since the proofs are very similar we omit details.

3. Sequential weights

The set of weights $W = \{w_1, w_2, \ldots, w_n\}$ is *sequential* if $w_n | w_{n-1} | w_{n-2} \ldots | w_1$, where '|' stands for 'divides'. Thus, for example, {24, 8, 8, 4, 4, 4, 2, 1} is sequential. It is evident that a subset of a sequential set is itself sequential.

Sequential weights have been considered, for example, by Hartman and Olmstead (1993) and others. Hartman and Olmstead (1993) consider the sequential knapsack problem:

$$\text{Maximize} \sum_{j \in N} p_j x_j$$

$$\text{Subject to} \sum_{j \in N} w_j x_j \leq q, x_j \in \{0,1\} \; \forall j$$

where p_j's, w_j's and q are all positive integers, and the weights w_j are sequential. They show that this problem can be solved in $O(n)$ time. A fortiori P_1 can be solved in $O(n)$ time if the set of weights is sequential. However, both the algorithm of Hartman and Olmstead (1993) and its proof are quite complicated. We show below that the greedy algorithm described in the previous section solves P_1 when the weights are sequential.

Lemma 5. *If the set of weights $\{w_1, w_2, \ldots, w_n\}$ is sequential then for each j, $1 \leq j < n$, either $w_j > w_{j+1} + w_{j+2} + \cdots + w_n$ or else we can find an index k, $j < k \leq n$, such that $w_j = w_{j+1} + \cdots + w_k$.*

Proof. Suppose the lemma to be false. Then we can find an index k, $j < k < n$, such that $w_j > w_{j+1} + \cdots + w_k$ and $w_j < w_{j+1} + \cdots + w_k + w_{k+1}$. Then $w_{k+1} > w_j - (w_{j+1} + \cdots + w_k)$. However, since w_{k+1} is a factor of each of $w_k, \ldots, w_j$, the left-hand side of the above inequality divides the right-hand side. However, then the right-hand side must be zero. So $w_j = w_{j+1} + \cdots + w_k$ which is a contradiction. □

Theorem 6. *The greedy algorithm solves Problem* P_1 *if the set of weights* $\{w_1, w_2, \ldots, w_n\}$ *is sequential.*

Proof. The proof will be by contradiction. Suppose that the set of weights $\{w_1, w_2, \ldots, w_n\}$ is sequential but the greedy solution g is not optimal to P_1. Let x then be an optimal solution to P_1 with the property that there exists an index i, $1 \le i \le n$, such that $g_j = x_j$, for $j < i$, $g_i \ne x_i$, and for any other optimal solution z to P_1, $g_j \ne z_j$, for some $j \le i$. In other words x is an optimal solution with the property that the least (i.e. left-most) index in which g differs from x is as large as possible. We will derive a contradiction by exhibiting another optimal solution y which agrees with g in at least its first i coordinates.

By Lemma 1 $g_i = 1$ and $x_i = 0$. Consider the set $S = \{w_j |\ j > i,\ x_j = 1\}$. Since x is optimal and g is not, $\Sigma_{j=1}\ w_j g_j < \Sigma_{j=1}\ w_j x_j$ which implies that $w_i < \Sigma_{j \in S}\ w_j$. Now: $\{i\} \cup S$ being a subset of W is itself sequential. Hence, $w_i = \Sigma_{j \in S'}\ w_j$ for some $S' \subseteq S$. It follows that from x we can obtain a solution y with same objective value as x by setting $y_i = 1$, $y_j = 0$ for all $j \in S'$, and $y_j = x_j$ otherwise. We have $g_j = y_j$, for $j \le i$. Which proves the theorem. □

The Greedy Algorithm is to be preferred to that of Hartman and Olmstead on several counts. It is simpler to describe, implement and validate, involves less computational overhead and has the same computational complexity. Most importantly it is possible to solve $P_1(i)$ for *every* i in $O(n)$ time with the help of the greedy solution.

Theorem 7. $P_1(i)$ *can be solved for every* i *in overall* $O(n)$ *time, if the set of weights* $\{w_1, w_2, \ldots, w_n\}$ *is sequential.*

Proof. Let R denote the optimal objective value of P_1. Suppose first that $g_i = 0$. Consider an execution of the greedy algorithm on the set of weights $W \backslash \{w_i\}$ and let y denote the vector output at the end. By using exactly the same argument as in the proof of Theorem 3 we can show that y is simply obtained from g by suppressing g_i and that the optimal objective values of P_1 and $P_1(i)$ are identical.

Suppose next that $g_i = 1$. Let $W_i = \{w_j |\ j > i,\ g_j = 0\}$. Since $\{i\} \cup W_i$ is sequential, it follows from Lemma 5 that there are exactly two distinct possibilities. Either $w_i > \Sigma_{j \in W_i}\ w_j = \Sigma_{j > i}\ w_j\ (1 - g_j)$ or else we can a find $V_i \subseteq W_i$ such that $w_i = \Sigma_{j \in V_i}\ w_j$. In the first case, the greedy algorithm, when run on the set of weights $W \backslash \{w_i\}$ outputs the solution $y = (y_1, y_2, \ldots, y_{i-1}, y_{i+1}, \ldots, y_n)$, where $y_j = g_j$, for all $j < i$, and $y_j = 1$, for all $j > i$. This is therefore the optimal solution to P_i. The optimal objective value of $P_1(i)$ in this case is $R - w_i + \Sigma_{j > i}\ w_j(1 - g_j)$.

In the remaining case the solution $z = (z_1, z_2, \ldots, z_{i-1}, z_{i+1}, \ldots, z_n)$, where $z_j = 1$, for all $j \in V_i$ and $z_j = g_j$ otherwise, has the objective value R. This must therefore be the optimal solution to $P_1(i)$ and the optimal objective values of P_1 and $P_1(i)$ must be identical in this case too.

Now running the greedy algorithm once to determine g requires $O(n)$ time. It is also possible to compute all the partial sums $\Sigma_{j>i}\ w_j(1-g_j)$, $1<j\leq n$, in $O(n)$ time recursively. It is clear from the above discussion that if g and these partial sums are available the optimal objective value of each $P_1(i)$ can be determined using a *constant* number of arithmetic operations. The result now follows. $\square$

We have not been able to find a polynomial time algorithm for determining the number of coalitions for which a player is pivotal when the set of weights W is sequential. Neither have we been able to prove that the problem is necessarily difficult. However, it is easy to solve P_2 if the set of weights W, in addition to being sequential, satisfy the following condition.

3.1. Dominance condition

Let $d_1>d_2>\ldots>d_r$ be the *distinct* values of weights $w_1, w_2, \ldots, w_n$ belonging to a sequential set. Then $d_k=m_k d_{k+1}$, where $m_k>1$, $\forall k$, $1\leq k<r$. Let $N_k=\{i|w_i=d_k\}$, and $n_k=|N_k|$. Then the dominance condition holds if $m_k>n_{k+1}$ $\forall k$, $1\leq k<r$.

Thus, for example, the set $\{48, 16, 16, 4, 4, 4, 2, 1\}$ is a sequential set of weights which satisfies the dominance condition. Here $d_1=48$, $d_2=16$, $d_3=4$, $d_4=2$, $d_5=1$ and $n_1=1$, $n_2=2$, $n_3=3$, $n_4=1$, $n_5=1$.

Lemma 8. *If the set of weights $\{w_1, w_2, \ldots, w_n\}$ is sequential and satisfies the dominance condition then $\forall j\in N_k$, $1\leq k<r$, $w_j>\Sigma\ \{w_p|p\in N_i,\ i>k\}$.*

Proof. If $k=r-1$ the lemma follows obviously from the dominance condition. Let us then suppose the lemma to be true for $k=i+1, \ldots, r-1$. Now let $k=i$. For all $j\in N_i$:

$$w_j=d_i=m_i\ d_{i+1}\geq(n_{i+1}+1)d_{i+1}=\sum_{p\in N_{i+1}} w_p+d_{i+1}$$

$>\Sigma\ \{w_p|p\in N_{i+1}\}+\Sigma\ \{w_p|p\in N_k,\ k>i+1\}$, by our hypothesis. Hence, for all $j\in N_i$, $w_j>\Sigma\ \{w_p|p\in N_k,\ k>i\}$. Hence the lemma is true for i. This concludes the proof by induction. $\square$

Suppose that W is a sequential set of weights satisfying the dominance condition. If x is any feasible solution to P_1 we will write $L_k(x)=\{i|i\in N_k,\ x_i=1\}$ and $l_k(x)=|L_k(x)|$.

Lemma 9. *Let x be any feasible solution to P_1, different from the greedy solution g, such that $l_k(g)=l_k(x)$, for $k<i$, and $l_i(g)\neq l_i(x)$, i being a given index between 1 and r. Then $l_i(g)>l_i(x)$.*

Proof. $l_i(x)>l_i(g)$ would imply that $\Sigma\ \{w_j g_j|j\in L_k(g),\ k\leq i\}<\Sigma\ \{w_j x_j|j\in L_k(x),\ k\leq i\}\leq q$. However, then the greedy algorithm must set $g_i=1$ for some element in $L_i(x)\backslash L_i(g)$ which is a contradiction. $\square$

Theorem 10. *Suppose that the set of weights* $\{w_1, w_2, \ldots, w_n\}$ *is sequential and satisfies the dominance condition. Let* $g = (g_1, g_2, \ldots, g_n)$, *denote the greedy solution to* P_1. *Then the number of feasible solutions to* P_1 *is*:

$$\left(\sum_{k=0}^{l_1(g)-1}\binom{n_1}{k}\right)2^{\sum_{i=2}^{r} n_j} + \sum_{i=2}^{r}\prod_{j=1}^{i-1}\binom{n_j}{l_j(g)}\left(\sum_{k=0}^{l_i(g)-1}\binom{n_i}{k}\right)2^{\sum_{j=i+1}^{r} n_j} + \prod_{j=1}^{r}\binom{n_j}{l_j(g)}$$

where $\binom{n}{r}$ *is the number of ways in which r objects may be chosen from n.*

Proof. Let S denote the set of feasible solutions to P_1. For $1 \le i \le r$, let $S_i = \{x | x$ is a feasible solution to P_1, $l_j(g) = l_j(x)$ for $j < i$, and $l_i(g) > l_i(x)\}$. Let $S_{r+1} = \{x | x$ is a feasible solution to P_1, $l_j(g) = l_j(x)\ \forall j, 1 \le j \le r\}$. Then from Lemma 9 it follows that S is the disjoint union of the sets S_i, $1 \le i \le r+1$, and so $|S| = \Sigma_1^{r+1} |S_i|$.

Now $|S_{r+1}|$ is clearly $\prod_{j=1}^{r}\binom{n_j}{l_j(g)}$ since setting $x_j = 1$, for any $l_j(g)$ of the n_j indices in N_j gives rise to a solution in S_{r+1}. We now calculate the number of solutions in S_i for any i, $1 \le i \le r$. To obtain a solution $x \in S_i$, $2 \le i \le r$ we have to set $x_p = 1$ for *exactly* $l_j(g)$ of the n_j indices p in N_j, $\forall j < i$. This can be done in $\binom{n_j}{l_i(g)}$ ways. We also have to set $x_p = 1$ for *at most* $l_i(g) - 1$ of the n_i indices p in N_i, $1 \le i \le r$. It is possible to choose these in $\Sigma_{k=0}^{l_i(g)-1}\binom{n_i}{k}$ ways. Suppose that we have obtained a partial solution x whose first $\Sigma_{j \le i}\ n_j$ coordinates have been fixed in this way. Using Lemma 8 we see that $\Sigma\ \{w_j | j \in N_k, k > i\} < d_i \le q - \Sigma\ \{w_j x_j | j \in N_k, k \le i\}$ so that any assignment of 0's and 1's to the last $\Sigma_{j>i}\ n_j$ coordinates yields a feasible solution to P_1. There are $2^{\Sigma_{j=i+1}^{r} n_j}$ such possible assignments. Hence:

$$|S_1| = \left(\sum_{k=0}^{l_1(g)-1}\binom{n_1}{k}\right)2^{\sum_{i=2}^{r} n_j} \text{ and } |S_i| = \prod_{j=1}^{i-1}\binom{n_j}{l_j(g)}\left(\sum_{k=0}^{l_i(g)-1}\binom{n_i}{k}\right)2^{\sum_{j=i+1}^{r} n_j}$$

if $i \ge 2$. The result now follows. □

Example. Let $W = \{48, 16, 16, 4, 4, 4, 2, 1\}$, $q = 75$. Then the greedy solution to P_1 is $g = (1, 1, 0, 1, 1, 0, 1, 1)$. So $l_1(g) = 1$, $l_2(g) = 1$, $l_3(g) = 2$, $l_4(g) = 1$, $l_5(g) = 1$. Applying Theorem 10, we see that the number of feasible solutions to P_1 is:

$$(1 \times 2^7) + (1 \times 1 \times 2^5) + (1 \times 2 \times 4 \times 2^2) + (1 \times 2 \times 3 \times 1 \times 2^1)$$
$$+ (1 \times 2 \times 3 \times 1 \times 1 \times 2^0) + (1 \times 2 \times 3 \times 1 \times 1) = 216$$

It follows in a straightforward way from Theorem 10 that P_2 can be solved in $O(n^2)$ time if the set of weights W is sequential and satisfies the dominance condition. Since any subset of such a W inherits these properties $P_2(i)$ can also be solved in $O(n^2)$ time for any given i.

4. Conclusion

In the present paper we have considered weighted majority games. We have shown that it is easy to determine whether or not a player is pivotal for some coalition if the weights are unbalanced, generalized unbalanced or sequential. We have also shown that it is easy to determine the number of coalitions for which a player is pivotal if the weights are unbalanced or generalized unbalanced. Finally we have shown that it is easy to determine the number of coalitions for which a player is pivotal if the weights are sequential and an additional dominance condition is satisfied.

We end by stating problems which we have not been able to resolve. Does there exist a polynomial time algorithm for determining the number of coalitions for which a given player is pivotal when the weights are sequential and the dominance condition does not obtain? Or is the problem provably difficult in this case? What other choices of weights make $P_1(i)$ and $P_2(i)$ easy?

References

Aho, A.V., Hopcroft, J.E., Ullman, J.D., 1974. The Design and Analysis of Computer Algorithms, Addison-Wesley.

Hartman, M., Olmstead, T., 1993. Solving sequential knapsack problems. Operations Research Letters 13, 225–232.

Konheim, A.G., 1981. Cryptography: A Primer, John Wiley.

Papadimitriou, C.H., 1994. Computational Complexity, Addison Wesley.

Prasad, K., Kelly, J.S., 1990. NP-completeness of some problems concerning voting games. International Journal of Game Theory 19, 1–9.

Roberts, F.S., 1976. Discrete Mathematical Models, Prentice-Hall, Englewood Cliffs, NJ.

Owen, G., 1995. Game Theory, Academic Press.