

Scheduling Jobs with a Common Due Date via Cooperative Game Theory

Irinel Dragan

University of Texas at Arlington, Mathematics, Arlington, USA

Email: dragan@uta.edu

Received October 18, 2012; revised December 30, 2012; accepted January 7, 2013

Copyright © 2013 Irinel Dragan. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

ABSTRACT

Efficient values from Game Theory are used, in order to find out a fair allocation for a scheduling game associated with the problem of scheduling jobs with a common due date. A four person game illustrates the basic ideas and the computational difficulties.

Keywords: Schedule; Efficient Value; Egalitarian Value; Egalitarian Nonseparable Contribution; Shapley Value; Cost Excesses; Lexicographic Ordering; Cost Least Square Prenucleolus

1. A Scheduling Game and Simple Solutions

A machine may process n jobs, J_1, J_2, \dots, J_n , with the completion times p_1, p_2, \dots, p_n , all positive numbers. No two jobs can be simultaneously done, and for all jobs there is a common due date d positive. Any schedule σ is a sequence of jobs, and no preemption is allowed. The schedule is determined by the numbers

$C_i(\sigma), \forall i \in N$, the completion dates of the jobs J_i in σ . Any deviation from the due date will be penalized, either an early or a late completion relative to the due date. The total time deviation in a schedule σ is given by

$$\Delta(\sigma) = \sum_{i \in N} |C_i(\sigma) - d|. \quad (1)$$

The usual scheduling problem is to: find out the schedule σ^* for which the total deviation is minimal. In [1], J. J. Kanet solved the problem for the case when the sum of completion times is smaller than, or equal to d , and gave an algorithm for computing a schedule with a minimal deviation. Of course, this algorithm may be used to find the total penalty for this schedule and also to find the total penalty for any minimal deviation corresponding to any subset of jobs. This makes sense in the case when the costs of the deviations, early or late, are proportional to their size. In the following, we assume that the costs are equal to the penalties. A more general case other than Kanet's has been solved by M. U. Ahmed and P. S. Sundararaghavan in [2]. In [3], N. G. Hall and M. E. Posner consider similar problems. The literature

connected to more general cases is huge, and the conclusions obtained in the present paper can be applied to most other cases. For the present discussion, the simplest case offered by Kanet's algorithm, with the above assumptions, is good enough to suggest similar approaches in all other cases, in connection with a new problem to be introduced below.

Assuming that the grand coalition has been formed and the total penalty for early and late deviation, $w(N)$, has been computed by some algorithm, a new problem is: how much should be a fair individual penalty for each job?

To answer the question, we now build the following cooperative game with transferable utilities: let

$N = \{1, 2, \dots, n\}$ be the set of players, the player i be the customer ordering the job J_i , for each

$i = 1, 2, \dots, n$ Consider any coalition of customers,

$S, S \subseteq N, S \neq \emptyset$, and notice that if $S = \{i\}$, then the

minimal schedule starts the corresponding job at $d - p_i$, and there will be no deviation from the due date. Therefore, if we denote the deviation for coalition S , by $w(S)$, we have $w(\{i\}) = 0$. An algorithm for computing the minimal deviation, for example Kanet's algorithm, will provide the total deviation $w(S) \geq 0$, when $|S| \geq 2$. In this way, we get a cooperative TU game (N, w) , in which we want to divide fairly $w(N)$.

To make the paper self contained let us sketch Kanet's algorithm which will be used in the example shown be-

low. Let S be any coalition and denote by B_S , (before S), a sequence of jobs which were already selected, with non-increasing processing times and the last job ending at d ; also, denote by A_S , (after S), another sequence of jobs which were already selected, with non-decreasing processing times and starting at d . Now, assume that we have $B_S \cap A_S = \emptyset, |B_S| = |A_S|$, Kanet's algorithm is continuing to build the partition of S as follows: if $|B_S| + |A_S| \leq |S| - 1$, select the non-selected element of S with a maximal processing time and take it as the last job in B_S , (now we have $|B_S| = |A_S| + 1$). Further, if with the new B_S , we have $|B_S| + |A_S| \leq |S| - 1$, then choose the non-selected job in S with a maximal processing time and take it as the first job in A_S , (that is starting at d). Repeat the procedure, selecting alternatively players in B_S , then in A_S , until S is exhausted; then, the time deviation is computed by formula (1) for the corresponding schedule and in the same way for any subset of players.

Example 1: Let $\{J_1, J_2, J_3, J_4\}$ be a set of jobs to be processed on a single machine, with the processing times $p_1 = 12, p_2 = 10, p_3 = 8, p_4 = 5$, and the due date is $d = 39$, such that the Kanet's condition shown above holds. We can compute for the set of players $N = \{1, 2, 3, 4\}$, and its subsets, the total penalties. Kanet's algorithm will generate the game

$$\begin{aligned} w(\{1\}) &= w(\{2\}) = w(\{3\}) = w(\{4\}) = 0, \\ w(\{1, 2\}) &= 10, w(\{1, 3\}) = w(\{2, 3\}) = 8, \\ w(\{1, 4\}) &= w(\{2, 4\}) = w(\{3, 4\}) = 5, \\ w(\{1, 2, 3\}) &= 18, w(\{1, 2, 4\}) = 15, \\ w(\{1, 3, 4\}) &= w(\{2, 3, 4\}) = 13, w(\{1, 2, 3, 4\}) = 28. \end{aligned}$$

This corresponds to the total deviations of all coalitions, and our problem is to: find out how we should divide fairly $w(N) = 28$ among the players? We start by showing two simple solutions: the Egalitarian allocation and the Egalitarian non-separable contribution allocation. Denoting the first by x^* , we get $x^* = (7, 7, 7, 7)$.

Denoting the second by y^* , which is given in general by formula

$$y_i^* = w(N) - w(N - \{i\}) + \frac{1}{n} \left\{ w(N) - \sum_{j \in N} [w(N) - w(N - \{j\})] \right\}, \forall i \in N, \tag{2}$$

we get the marginal contributions $M_j = w(N) - w(N - \{j\}), \forall j \in N$, equal to 15, 15, 13, 10, so that the sum makes 53, and from each marginal contribution we should subtract 25/4, to obtain

$$y^* = \left(\frac{35}{4}, \frac{35}{4}, \frac{27}{4}, \frac{15}{4} \right).$$

Looking at the characteristic values of our game, shown in example 1, we see that the players 1 and 2 seem to be equal, while players 3 and 4 are weaker, hence the last two should be asked to pay smaller individual penalties.

The first solution does not seem to show this, while the second seems more fair, we shall see a method below to compare the fairness of the solutions.

2. Individual Penalties: Set Solutions, Shapley Value

To evaluate the fairness of a possible solution z , we may use the excess functions; however, here it seems more appropriate to use some similar functions that we shall call the "cost excess" functions. For any coalition $S, S \subseteq N, S \neq \emptyset$, and any allocation z , the cost excess function is

$$\varepsilon(S, z) = \sum_{i \in S} z_i - w(S). \tag{3}$$

These are the negatives of usual excess functions, obviously, we have $2^n - 2$ such functions, because for the grand coalition, for any allocation, by definition we have $\varepsilon(N, z) = 0$. In words, the cost excess is the difference between what the coalition S will pay in z to contribute as close as possible to the total penalty for herself, while it is also contributing to the total penalty for N . Then, what we want to do is to choose the allocation z which minimizes all cost excesses on the set of allocations. Note that the sum of all cost excesses is a constant, because for any allocation z we have

$$\sum_{S \subseteq N} \varepsilon(S, z) = 2^{n-1} w(N) - \sum_{S \subseteq N} w(S). \tag{4}$$

Moreover, we can define the average cost excess

$$\varepsilon(w) = \frac{1}{2^n - 1} \sum_{S \subseteq N} \varepsilon(S, z), \tag{5}$$

which by formula (4) does not depend on the allocation z . This means that if the allocation of some coalition is increased, then the allocation of at least one other coalition will be decreased. How the cost excesses are used to compare the fairness of two allocations is illustrated below.

Example 2: Return to example 1, and write the cost excesses for that game (N, w) , and any allocation

$$\begin{aligned} \varepsilon(\{1\}, z) &= z_1, \varepsilon(\{2\}, z) = z_2, \\ \varepsilon(\{3\}, z) &= z_3, \varepsilon(\{4\}, z) = z_4, \\ \varepsilon(\{1, 2\}, z) &= z_1 + z_2 - 10, \\ \varepsilon(\{1, 3\}, z) &= z_1 + z_3 - 8, \end{aligned}$$

$$\begin{aligned} \varepsilon(\{1,4\}, z) &= z_1 + z_4 - 5, \varepsilon(\{2,3\}, z) = z_2 + z_3 - 8, \\ \varepsilon(\{2,4\}, z) &= z_2 + z_4 - 5, \varepsilon(\{3,4\}, z) = z_3 + z_4 - 5, \\ \varepsilon(\{1,2,3\}, z) &= z_1 + z_2 + z_3 - 18, \\ \varepsilon(\{1,2,4\}, z) &= z_1 + z_2 + z_4 - 15, \\ \varepsilon(\{1,3,4\}, z) &= z_1 + z_3 + z_4 - 13, \\ \varepsilon(\{2,3,4\}, z) &= z_2 + z_3 + z_4 - 13. \end{aligned}$$

Our problem is to minimize all cost excesses, while we are on the set of allocations, that is the efficiency condition holds. In other words, we want to minimize the maximal cost excess, subject to efficiency condition, or to use another method to solve a multi objective linear programming problem.

Let us try to evaluate the fairness of the two allocations offered until now.

For the Egalitarian solution, we can compute the cost excesses and put them in a vector of non increasing excesses, which may be called the vector of unhappiness, as the components are taken in the order of non increasing unhappiness

$$\theta(x^*) = (9, 9, 9, 8, 8, 7, 7, 7, 7, 6, 6, 6, 4, 3).$$

It follows that the most unhappy coalitions are the two person coalitions in which one of the players is player 4. For the Egalitarian non separable contribution, we find the vector of unhappiness

$$\theta(y^*) = \left(\frac{35}{4}, \frac{35}{4}, \frac{15}{2}, \frac{15}{2}, \frac{15}{2}, \frac{15}{2}, \frac{15}{2}, \frac{27}{4}, \frac{25}{4}, \frac{25}{4}, \frac{25}{4}, \frac{11}{2}, \frac{15}{4} \right),$$

and the most unhappy coalitions are $\{1\}$ and $\{2\}$.

Moreover, we have also $\theta_1(x^*)$ larger than $\theta_1(y^*)$, that is the most unhappy coalition in x^* is more unhappy than the most unhappy coalition in y^* . We can say that y^* is better than x^* , or more fair. Note that the same thing could be said if some pairs of corresponding components in the two unhappiness vectors are equal, but the first one which is different is smaller in $\theta(y^*)$ than in $\theta(x^*)$. In this case, we also write $\theta(y^*) \leq_L \theta(x^*)$, where L means the lexicographic order, and read y^* is better than x^* . Until now, we have seen two simple solutions belonging to Game Theory, they are one point solutions because each one is providing a unique solution.

One of the set solutions from Game Theory is the CORE, which for a cost game (N, w) like ours is the set

$$CO(N, w) = \{z \in R^n : \varepsilon(N, z) = 0, \varepsilon(S, z) \geq 0, \forall S \subset N, S \neq \emptyset\}. \tag{6}$$

Any element of the CORE is considered as a good allocation, because such an allocation covers the total penalty for each coalition. Looking at the two simple solutions of example 1, which as seen in example 2 have all excesses non negative, we see that both are in the CORE, but, of course, there are others with the same property. Moreover, we can also see that the sum of excesses equals 96, that is the worth obtained in formula (4) for $n = 4$.

The most famous one point solution, which may also be in the CORE, is the Shapley Value, introduced in [4], and defined by a set of axioms, describing some basic properties required for a fair solution. The Shapley Value was proved there to be given by the formula

$$SH_i(N, w) = \sum_{S: i \in S} \frac{(s-1)!(n-s)!}{n!} [v(S) - v(S - \{i\})], \forall i \in N. \tag{7}$$

Example 3: For the game considered in example 1, we get

$$SH(N, w) = (8, 8, 7, 5).$$

Computing the cost excesses and ordering them, we obtain

$$\theta(SH) = (8, 8, 8, 8, 7, 7, 7, 7, 7, 6, 6, 5, 5).$$

We get $\theta(SH) \leq_L \theta(y^*) \leq_L \theta(x^*)$, because the first components of the unhappiness vectors are in this order, hence the Shapley Value is better than the Egalitarian non separable contribution, which is better than the Egalitarian solution. Note that this may not be the case for other games. Note also that if the game is large, then the Shapley Value may not be easy to compute. An algorithm based upon the so called Average per capita formula, given by the author in [5], may be used, as it will be explained in the next section and the algorithm is allowing even a parallel computation of the Shapley Value. Similar situations may occur in connection with the other values.

3. The Cost Least Square Prenucleolus

In the following, we may consider as a solution the Least Square Prenucleolus of the game, introduced by L. Ruiz, F. Valenciano and J. Zarzuelo in [6]. This is similar to the Prenucleolus, introduced in connection with the Nucleolus, due to D. Schmeidler [7], except that this was defined by means of the following quadratic programming problem

$$\text{Minimize } f(w, z) = \sum_{S \in N} [\varepsilon(S, z) - \varepsilon(w)]^2, \tag{8}$$

subject to

$$\varepsilon(N, z) = 0. \tag{9}$$

By using the Kuhn-Tucker conditions, in (8), (9), it has been shown that our problem has a unique solution, that the authors called the Least Square Prenucleolus, namely

$$LS_i(N, w) = \frac{w(N)}{n} + \frac{1}{\alpha n} \left[na_i(w) - \sum_{j \in N} a_j(w) \right], \forall i \in N, \tag{10}$$

where

$$a_i(w) = \sum_{S: i \in S} w(S), \forall i \in N, \alpha = \sum_{s=1}^{n-1} \binom{n-2}{s-1}. \tag{11}$$

As the cost excesses are replacing the excesses, we called this value the Cost Least Square Prenucleolus, but the expressions (10) and (11) are the same even in our case.

Example 4: Computing by (10) and (11) this solution for our game, we get

$$LS(N, w) = \left(\frac{129}{16}, \frac{129}{16}, \frac{113}{16}, \frac{77}{16} \right).$$

The vector of unhappiness is $\theta(LS^*)$ (see foot-note).

Notice that the sum of cost excesses is also 96. Now, checking the comparison of the new solution with the other three solutions, we obtain

$$SH(N, w) \leq_L LS(N, w) \leq_L y^* \leq x_L^*,$$

that is the Shapley Value seems to be more fair than the other three values. Of course, the Schmeidler's Prenucleolus may also be computed; the computational method, due to A. Kopelowitz [8], is also shown in [9]. However, this includes a long computation for solving a sequence of linear optimization problems. The Prenucleolus would be the best, by the definition of the value.

Another principle may be used to choose the appropriate allocation: for each allocation available, compute the difference between the cost for the most unhappy coalition and the happiest coalition and choose the allocation that gives the smallest difference. Such an allocation would not give a high difference of costs between the happy and the unhappy coalitions. In our case, we have

$$d_{SH} = 3, d_{LS} = \frac{129}{16} - \frac{77}{16} = \frac{13}{4}, d_{y^*} = 5, d_{x^*} = 6.$$

Notice that by the last principle the four values are ordered in the same way.

4. Conclusions

The technology put together in the present paper applies

to other scheduling problems in which the associated scheduling game can still be generated by some algorithm. Some of the following remarks may help:

1) The above discussion was illustrated by examples 1, 2, 3, 4 relative to a four person game. If we have $n \geq 5$ jobs, under the same conditions like above, Kanet's algorithm still applies when the Kanet's assumption holds. If the objective is different, for example to minimize a weighted combination of deviations relative to a common due date, then the algorithm by Hall and Posner should be used to get the scheduling game.

2) As soon as the game is available, the problem of dividing fairly the worth of the grand coalition is the problem of choosing an efficient value from Game Theory, for which the computation could be done. As all singletons have a zero worth, the Center of the imputation set [9] becomes the Egalitarian value, which in general is not fair. The Egalitarian non-separable contribution may be an alternative allocation. The Shapley Value, which has a lot of properties derived from the axioms, is the most preferred by almost all scientists, but for more than ten players it becomes difficult to compute. For such large games it may be better to use the formula given by the author in [5], called the Average per capita formula

$$SH_i(N, w) = \sum_{s=1}^{s=n} \frac{w_s - w_s^i}{s}, \forall i \in N, \tag{12}$$

where w_s is the average worth of coalitions of size s , and w_s^i is the average worth of coalitions of size s , that do not contain player i , with $w_n^i = 0, \forall i \in N$. It is obvious that the task can be performed by n teams, and each one computes one ratio for one s .

3) In the computation of the Cost Nucleolus by Kopelowitz' method [8,9] the passage from one LP problem to the next is not described in details in most sources. The complementary conditions show which cost excesses should remain constant on all optimal solutions of the current problem, and should be kept constant in the next problem. These equations should replace the corresponding inequalities of the current problem and remain satisfied until we meet a problem which has a unique solution. The solution of the quadratic programming problem seems easier to compute.

The generalized nucleolus may be used as a solution of any Multicriteria Linear Programming problem, as shown by the author in [10], working with a three person game. The basic idea appeared in a former paper of the author [11], as well as in the more recent paper by E. Marchi and J. A. Oviedo [12].

$$\theta(LS^*) = \left(\frac{129}{16}, \frac{129}{16}, \frac{63}{8}, \frac{63}{8}, \frac{57}{8}, \frac{57}{8}, \frac{113}{16}, \frac{111}{16}, \frac{111}{16}, \frac{55}{8}, \frac{49}{8}, \frac{95}{16}, \frac{83}{16}, \frac{77}{16} \right).$$

REFERENCES

- [1] J. J. Kanet, "Minimizing the Average Deviation of Job Completion Times about a Common Due Date," *Naval Research Logistics Quarterly*, Vol. 28, No. 4, 1981, pp. 643-652. [doi:10.1002/nav.3800280411](https://doi.org/10.1002/nav.3800280411)
- [2] M. U. Ahmed and P. S. Sundararaghavan, "Minimizing the Weighted Sum of Late and Early Completion Penalties in a Single Machine," *IEEE Transactions*, Vol. 22, No. 3, 1990, pp. 288-290. [doi:10.1080/07408179008964183](https://doi.org/10.1080/07408179008964183)
- [3] N. G. Hall and M. E. Posner, "Earliness-Tardiness Scheduling Problems, I: Weighted Deviation of the Completion Times about a Common Due Date," *Operations Research*, Vol. 39, No. 5, 1991, pp. 839-846.
- [4] L. S. Shapley, "A Value for n -Person Games," *Annals of Mathematics*, Vol. 28, 1953, pp. 307-317.
- [5] I. Dragan, "An Average per Capita Formula for the Shapley Value," *Libertas Mathematica*, Vol. 12, 1992, pp. 139-146.
- [6] L. Ruiz, F. Valenciano and J. Zarzuelo, "The Least Square Prenucleolus and the Least Square Nucleolus, Two Values for TU Games Based on the Excess Vector," *International Journal of Game Theory*, Vol. 25, No. 1, 1996, pp. 113-134.
- [7] D. Schmeidler, "The Nucleolus of a Characteristic Function Game," *SIAM Journal on Applied Mathematics*, Vol. 17, No. 6, 1967, pp. 1163-1170. [doi:10.1137/0117107](https://doi.org/10.1137/0117107)
- [8] A. Kopelowitz, "Computation of the Kernels of Simple Games and the Nucleolus of n -Person Game," RM 31, Hebrew University of Jerusalem, Jerusalem, 1967.
- [9] G. Owen, "Game Theory," 3rd Edition, Academic Press, New York, 1995.
- [10] I. Dragan, "A Game Theoretic Approach for Solving Multiobjective Linear Programming Problems," *Libertas Mathematica*, Vol. 30, 2010, pp. 149-158.
- [11] I. Dragan, "A Game Theoretic Approach for Solving Multiobjective Linear Programming Problems: An Application to a Traffic Problem," *Quaderni dei Gruppi di Ricerca CNR*, Pisa, 1981
- [12] E. Marchi and J. A. Oviedo, "Lexicographic Optimality in the Multiple Objective Linear Programming: The Nucleolar Solution," *European Journal of Operational Research*, Vol. 57, No. 3, 1992, pp. 353-359. [doi:10.1016/0377-2217\(92\)90347-C](https://doi.org/10.1016/0377-2217(92)90347-C)