



HAL
open science

Generalized Nash Fairness solutions for Bi-Objective Minimization Problems

Minh Hieu Nguyen, Mourad Baiou, Viet Hung Nguyen, Thi Quynh Trang Vo

► **To cite this version:**

Minh Hieu Nguyen, Mourad Baiou, Viet Hung Nguyen, Thi Quynh Trang Vo. Generalized Nash Fairness solutions for Bi-Objective Minimization Problems. 2022. hal-03890300v2

HAL Id: hal-03890300

<https://hal.science/hal-03890300v2>

Preprint submitted on 15 Dec 2022 (v2), last revised 19 Oct 2023 (v3)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Generalized Nash Fairness solutions for Bi-Objective Minimization Problems

Minh Hieu Nguyen¹ | Mourad Baiou¹ | Viet Hung Nguyen¹ | Thi Quynh Trang Vo¹

¹INP Clermont Auvergne, Univ Clermont Auvergne, Mines Saint-Etienne, CNRS, UMR 6158 LIMOS, 1 Rue de la Chebarde, Aubiere Cedex, France

Correspondence

Viet Hung Nguyen

Email: viet_hung.nguyen@uca.fr

In this paper, we consider a special case of Bi-Objective Optimization (BOO), called *Bi-Objective Minimization* (BOM), where two objective functions to be minimized take only positive values. As well as for BOO, most methods proposed in the literature for solving BOM focus on computing the Pareto-optimal solutions that represent different trade-offs between two objectives. However, it may be difficult for a central decision-maker to determine the preferred solutions due to a huge number of solutions in the Pareto set. We propose a novel criterion for selecting the preferred Pareto-optimal solutions by introducing the concept of ρ -Nash Fairness (ρ -NF) solutions inspired from the definition of proportional fairness. The ρ -NF solutions are the Pareto-optimal solutions achieving some proportional Nash equilibrium between the two objectives. The positive parameter ρ is introduced to reflect the relative importance of the first objective to the second one. For this work, we will discuss some existential and algorithmic questions about the ρ -NF solutions by first showing their existence for BOM. Furthermore, the set of ρ -NF solutions can be a strict subset of the Pareto set. As there are possibly many ρ -NF solutions, we focus on extreme ρ -NF solutions achieving the smallest values for one of the objectives. Then, we propose

two Newton-based iterative algorithms for finding extreme ρ -NF solutions. Finally, we present computational results on some instances of the Bi-Objective Travelling Salesman Problem and the Bi-Objective Shortest Path Problem.

KEYWORDS

Bi-Objective Optimization, Bi-Criteria Decision Making, Pareto optimal, Weighted Sum Method, Proportional Fairness, Bi-Objective Travelling Salesman Problem, Bi-Objective Shortest Path Problem

1 | INTRODUCTION

In this paper, we consider a special case of Bi-Objective Optimization (BOO), called *Bi-Objective Minimization* (BOM), where two objective functions to be minimized take only positive values. Many applications in telecommunications, logistics, economics, etc. can be formulated as BOM. We can list here several examples like the Bi-Objective Spanning Tree Problem [10] which aims at finding a spanning tree minimizing simultaneously the total cost and the diameter of the tree and the Bi-Objective Shortest Path Problem [33] where each arc is associated with a cost and a travel time and one desires to compute a shortest path minimizing the total cost and the total travel time between a given source to a given destination. In a recent conference paper [24], we considered a bi-objective version of the Travelling Salesman Problem where, in addition of the total cost, the difference of the maximum edge cost and the minimum edge cost in the tour is also to be minimized. This problem is a BOM which puts together the objective of the classic TSP and the one of the Balanced TSP introduced in [20]. In this paper, we present a generalized and extended version of [24] to BOM.

Obviously, popular methods for solving BOO can also be applied for solving BOM. Based on the concept of the Pareto-optimal solutions that are non-dominated with respect to each other, such methods usually construct a representation of the Pareto set that represents different trade-offs between the objectives [22]. They can be mainly divided into two classes, i.e. methods with a posterior articulation of preferences and methods with a priori articulation of preferences. In the former methods such as the Normal Boundary Intersection (NBI) [6], the Non-dominated Sorting Genetic Algorithm-II (NSGA-II) [7], a central decision-maker (CDM) selects manually her own preferred solutions from the Pareto set. Although these methods can provide all the Pareto-optimal solutions, it may be a difficult task due to a huge number of solutions in the Pareto set [29].

In practice, methods with a priori articulation of preferences [17], which consider the preferences of the CDM before running the optimization algorithm and then allow the algorithm to determine the solutions that reflect such preferences, have been used more extensively due to their computational efficiency. They usually formulate a single-objective optimization problem whose optimal solutions are the Pareto-optimal solutions to the bi-objective optimization problem. For instance, Weighted Sum Method [5], [36] scalarizes two objectives into a single objective by multiplying each objective with a weight supplied by the CDM. ϵ -constraints Method [15] keeps only one objective and use a CDM-specified value as an upper bound for the other objective. By changing this value, we are able to obtain some different optimal solutions. In the lexicographic method [11], the CDM is asked to establish a pre-defined order amongst the objective functions. Then this method solves a sequence of single-objective optimization problems, where each objective is optimized at a single point of time. In the goal programming [2], [3], the CDM determines an

aspiration level of the objective function. All the parameters of objective functions are assigned a goal value to be accomplished. Then the objective of goal-programming is to minimize the deviations from the set goals. There are also many other methods such as Sen's Multi-Objective Programming [31], Hypervolume/Chebyshev scalarization [13], achievement scalarizing problems of Wierzbicki [34], etc. However, most above approaches for BOM simply outline the methods and show that they can provide efficiently the Pareto-optimal solutions. Essentially, how to determine selection criterion for the preferred solutions and what is the signification of parameters using for the preferences remain the challenging questions.

In this paper, we propose a novel selection criterion for BOM which can guide efficiently the Weighted Sum Method to find the preferred Pareto-optimal solutions achieving some proportional Nash equilibrium between the two objectives in the context of fair competition based on proportional fairness [1], [19], [25]. The latter aims to provide a compromise between the utilitarian rule - which emphasizes overall system efficiency, and the egalitarian rule - which emphasizes individual fairness. In the context of BOM, proportional fairness means that the sum of proportional changes in objectives' value when switching from a preferred solution to any other feasible solutions is not negative. For our purpose, we consider a more general version of proportional fairness in order to take into account the relative importance of one objective to the other according to the point of view of the CDM. More precisely, we introduce the concept of the *generalized Nash Fairness* solution, i.e. ρ -*NF* solution, for BOM where $\rho > 0$ is a factor denoting the relative importance of the first objective comparing to the second one. This allows the CDM to consider that ρ percent change of the first objective is comparably equivalent to one percent change of the second one. Hence, when switching from a ρ -*NF* solution to any other feasible solutions, the sum of the factor ρ of the proportional change in the first objective and the proportional change in the second objective is not negative. In this paper, we present existential and algorithmic results for ρ -*NF* solutions of BOM problems.

The paper is organized as follows. In Section 2, we first show that for BOM, ρ -*NF* solutions always exist and moreover, the set of ρ -*NF* solutions is a subset of the Pareto set. We exhibit an example of BOM for which this inclusion can be strict. In Section 3, we address the question of how to algorithmically find ρ -*NF* solutions. For this, we focus on extreme ρ -*NF* solutions which are the ones with the smallest values for one of two objectives. We propose Newton-based iterative algorithms assuring the convergence to extreme ρ -*NF* solutions in a finite number of iterations. In Section 4, we discuss about the effect of ρ on the values of corresponding extreme ρ -*NF* solutions. Finally, in Section 5 we present computational results for the Bi-Objective Travelling Salesman Problem (BOTSP) introduced in [24] and the Bi-Objective Shortest Path Problem (BOSPP) [33].

Note that in two recent conference papers [23] and [24], we introduced the notion of *NF* solution which is a special case of ρ -*NF* solution when $\rho = 1$ for the Bi-Objective Assignment Problem (BOAP) and the BOTSP. In this paper, we introduce the concept of ρ -*NF* solution and generalize the theory to BOM.

2 | GENERALIZED NASH FAIRNESS SOLUTION

2.1 | Characterization of generalized NF solution

BOM can be generally formulated as

$$\min_{x \in \mathcal{X}} (P(x), Q(x))$$

where P, Q represent two objective functions, $\mathcal{X} \subseteq \mathbb{R}^d$ is a feasible set of decision vectors in d -dimensional application domain and $P(x), Q(x) > 0, \forall x \in \mathcal{X}$. Note that the feasible set \mathcal{X} is defined by some constraints. In this work, we

assume that X is finite. This hypothesis is verified for most practical examples of BOM, especially for the bi-objective combinatorial optimization problems.

As a special case of BOO, popular methods for solving BOM provide the Pareto-optimal solutions that represent different trade-offs between two objectives. However, in practice, it may be difficult for the CDM to determine a clear selection criterion for accurate a priori articulation of preferences in a representation of the Pareto set. In this section, we propose a novel criterion for selecting the preferred Pareto-optimal solutions. For this purpose, we introduce the concept of *generalized Nash Fairness* solution, i.e. ρ -*NF* solution, inspired from the definition of proportional fairness [1], [19], [25] where $\rho > 0$ reflects the relative importance between two objectives.

Let $(P, Q) = (P(x), Q(x))$ denote the solution corresponding to a feasible decision vector $x \in X$. In the standard definition of proportional fairness, an alternative assigning the greater values of two objectives is preferred, i.e. we have two utility functions to be maximized. However, by contrast in BOM, we prefer the alternative assigning the smaller values for P and Q , i.e. P, Q are two cost functions to be minimized. Furthermore, if $\rho > 0$ is a factor denoting the relative importance of P to Q , ρ percent change of P is now comparably equivalent to one percent change of Q . Thus, the ρ -*NF* solution should be such that, when compared to any other feasible solutions, the sum of the factor ρ of the proportional change of P and the proportional change of Q is not negative. Notice that it is easy for the CDM to explain the ρ -*NF* solutions using the signification of proportional fairness.

Let S be the set of all feasible solutions (P, Q) . Since X is finite, S is also finite. If $(P^*, Q^*) \in S$ is a ρ -*NF* solution for BOM, we have

$$\rho \frac{P - P^*}{P^*} + \frac{Q - Q^*}{Q^*} \geq 0, \forall (P, Q) \in S, \quad (1)$$

which is equivalent to

$$\rho \frac{P}{P^*} + \frac{Q}{Q^*} \geq \rho + 1, \forall (P, Q) \in S,$$

Hence, the ρ -*NF* solution for BOM can be defined as follows.

Definition 1 [*ρ -NF solution*] For a given value $\rho > 0$, $(P^*, Q^*) \in S$ is a ρ -*NF* solution for BOM if and only if $\rho \frac{P}{P^*} + \frac{Q}{Q^*} \geq \rho + 1, \forall (P, Q) \in S$.

The ρ -*NF* solution is a generalized concept of the *NF* solution that we have introduced in our conference papers [23] and [24]. In the following, we will discuss some existential questions about ρ -*NF* solution.

2.2 | Existence of ρ -*NF* solution

In case $\rho = 1$, it is known that there always exists a ρ -*NF* solution which can be obtained by minimizing the product of P and Q [23], [24]. Generally, we show the existence of a ρ -*NF* solution for every BOM.

Theorem 1 For all $\rho > 0$, there always exists a ρ -*NF* solution for BOM.

Proof We need to show that there exists a feasible solution $(P^*, Q^*) \in S$ such that

$$\rho \frac{P}{P^*} + \frac{Q}{Q^*} \geq \rho + 1, \forall (P, Q) \in S,$$

which is equivalent to

$$k_1 \frac{P}{P^*} + k_2 \frac{Q}{Q^*} \geq 1, \forall (P, Q) \in S,$$

where $k_1 = \frac{\rho}{\rho+1}$ and $k_2 = \frac{1}{\rho+1}$.

Since S is finite, there always exists a feasible solution $(P^*, Q^*) \in S$ such that

$$(P^*, Q^*) = \arg \min_{(P, Q) \in S} P^{k_1} Q^{k_2},$$

Let us show that (P^*, Q^*) is a ρ -NF solution. $\forall (P, Q) \in S$ we have $P^{k_1} Q^{k_2} \geq P^{*k_1} Q^{*k_2} > 0$. Since $\rho > 0$, $k_1, k_2 > 0$ and $k_1 + k_2 = 1$. Using the generalization of Young's inequality for products [35] (see Appendix 1), we get

$$k_1 \frac{P}{P^*} + k_2 \frac{Q}{Q^*} \geq \left(\frac{P}{P^*}\right)^{k_1} \left(\frac{Q}{Q^*}\right)^{k_2} = \frac{P^{k_1} Q^{k_2}}{P^{*k_1} Q^{*k_2}} \geq 1,$$

Hence, (P^*, Q^*) is a ρ -NF solution. □

Theorem 1 proves the existence of one ρ -NF solution for BOM that minimizes $P^{k_1} Q^{k_2}$, or equivalently minimizes $(k_1 \log P + k_2 \log Q)$. However, finding such a solution may be difficult as it requires to minimize a concave function. In the following, we show that each ρ -NF solution can be found by minimizing an appropriate strict convex combination of ρP and Q . More precisely, each ρ -NF solution can be obtained by solving the following optimization problem

$$\mathcal{P}_\rho(\alpha, \beta) = \min_{(P, Q) \in S} \rho\alpha P + \beta Q,$$

where $(\alpha, \beta) \in C := \{(\alpha, \beta) \mid \alpha, \beta > 0 \text{ and } \alpha + \beta = 1\}$, i.e. the set of all strict convex combinations defined by α and β , are the coefficients to be determined. Notice that in this paper, we assume the existence of the algorithms for solving the problem $\mathcal{P}_\rho(\alpha, \beta)$ with $(\alpha, \beta) \in C$, the problem minimizing P and the problem minimizing Q . For example, when P and Q are linear functions, $\rho\alpha P + \beta Q$ is also linear. In this case, we can suppose that we know how to optimize a linear function over X . The BOTSP and the BOSPP that we consider in the "computational results" section belong to this category. In addition, the solution of $\mathcal{P}_\rho(\alpha, \beta)$ will be characterized only by the solution (P, Q) and not by the decision vector of the solution. Thus, two solutions having the same value of (P, Q) will be considered as the same.

Theorem 2 $(P^*, Q^*) \in S$ is a ρ -NF solution if and only if (P^*, Q^*) is an optimal solution of $\mathcal{P}_\rho(\alpha^*, \beta^*)$ with $\alpha^* = \frac{Q^*}{P^* + Q^*}$ and $\beta^* = \frac{P^*}{P^* + Q^*}$.

Proof \implies Let (P^*, Q^*) be a ρ -NF solution. Let $\alpha^* = \frac{Q^*}{P^* + Q^*}$ and $\beta^* = \frac{P^*}{P^* + Q^*}$. Hence, $(\alpha^*, \beta^*) \in C$. We will show that (P^*, Q^*) is an optimal solution of $\mathcal{P}_\rho(\alpha^*, \beta^*)$.

As (P^*, Q^*) is a ρ -NF solution, we have

$$\rho \frac{P}{P^*} + \frac{Q}{Q^*} \geq \rho + 1, \forall (P, Q) \in S, \quad (2)$$

Multiplying (2) by $\beta^* Q^* > 0$ gives

$$\rho \frac{\beta^* Q^*}{P^*} P + \beta^* Q \geq \rho \beta^* Q^* + \beta^* Q^*, \forall (P, Q) \in S, \quad (3)$$

From the values of α^* and β^* , we have $\alpha^*P^* = \beta^*Q^*$. Thus, we deduce from (3)

$$\rho\alpha^*P + \beta^*Q \geq \rho\alpha^*P^* + \beta^*Q^*, \forall (P, Q) \in S,$$

Hence, (P^*, Q^*) is an optimal solution of $\mathcal{P}_\rho(\alpha^*, \beta^*)$.

\Leftarrow Now suppose that (P^*, Q^*) is an optimal solution of $\mathcal{P}_\rho(\alpha^*, \beta^*)$ with $\alpha^* = \frac{Q^*}{P^*+Q^*}$ and $\beta^* = \frac{P^*}{P^*+Q^*}$, we show that (P^*, Q^*) is a ρ -NF solution.

As (P^*, Q^*) is an optimal solution of $\mathcal{P}_\rho(\alpha^*, \beta^*)$, we have

$$\rho\alpha^*P + \beta^*Q \geq \rho\alpha^*P^* + \beta^*Q^*, \forall (P, Q) \in S,$$

Replacing α^* by $\frac{Q^*}{P^*+Q^*}$ and β^* by $\frac{P^*}{P^*+Q^*}$, we obtain

$$\rho \frac{P}{P^*} + \frac{Q}{Q^*} \geq \rho + 1, \forall (P, Q) \in S,$$

That means (P^*, Q^*) is a ρ -NF solution. \square

Theorem 2 states a necessary and sufficient condition for the ρ -NF solutions. We are interested now in the following question: Given a feasible solution $(P', Q') \in S$, how to assert that (P', Q') is a ρ -NF solution? We give the answer to this question in the next proposition.

Proposition 3 Given a feasible solution $(P', Q') \in S$. Let $\alpha' = \frac{Q'}{P'+Q'}$, $\beta' = \frac{P'}{P'+Q'}$ and (P^*, Q^*) be an optimal solution of $\mathcal{P}_\rho(\alpha', \beta')$, then (P', Q') is a ρ -NF solution $\iff \rho\alpha'P^* + \beta'Q^* - \rho\alpha'P' - \beta'Q' = 0$.

Proof \implies If $\rho\alpha'P^* + \beta'Q^* - \rho\alpha'P' - \beta'Q' = 0$ then (P', Q') is also an optimal solution of $\mathcal{P}_\rho(\alpha', \beta')$. Since $(\alpha', \beta') \in C$, $\alpha' = \frac{Q'}{P'+Q'}$ and $\beta' = \frac{P'}{P'+Q'}$, (P', Q') is a ρ -NF solution due to Theorem 2.

\Leftarrow If (P', Q') is a ρ -NF solution then (P', Q') is also an optimal solution of $\mathcal{P}_\rho(\alpha', \beta')$ due to Theorem 2. Thus, $\rho\alpha'P^* + \beta'Q^* = \rho\alpha'P' + \beta'Q'$ which leads to $\rho\alpha'P^* + \beta'Q^* - \rho\alpha'P' - \beta'Q' = 0$. \square

Note that BOM is a special case of multi-objective optimization where the concept of the Pareto-optimal (non-dominated, non-inferior) solution is widely applied for describing the set of efficient solutions [22]. For two feasible solutions $(P', Q'), (P'', Q'') \in S$, let us denote $(P', Q') < (P'', Q'')$, i.e. (P'', Q'') is strictly dominated by (P', Q') , if $P' \leq P'', Q' \leq Q''$ and $(P', Q') \neq (P'', Q'')$, i.e. we do not have $P' = P''$ and $Q' = Q''$ simultaneously. We recall the definition of the Pareto-optimal solution for BOM.

Definition 2 [Pareto-optimal solution] $(P', Q') \in S$ is a Pareto-optimal solution for BOM if and only if it is not strictly dominated by any other feasible solutions in S . In other words, there does not exist $(P'', Q'') \in S$ such that $(P'', Q'') < (P', Q')$.

In the following, we will prove that each ρ -NF solution is necessarily a Pareto-optimal solution.

Proposition 4 Each ρ -NF solution is necessarily a Pareto-optimal solution.

Proof Let $(P^*, Q^*) \in S$ be an optimal solution of $\mathcal{P}_\rho(\alpha^*, \beta^*)$ where $(\alpha^*, \beta^*) \in C$. We first show that (P^*, Q^*) is a Pareto-optimal solution.

Suppose that there exists a feasible solution $(P', Q') \in S$ satisfying $(P', Q') < (P^*, Q^*)$. Since $(\alpha^*, \beta^*) \in C$, $\rho > 0$ and $(P', Q') < (P^*, Q^*)$, we have $\rho\alpha^*P' + \beta^*Q' < \rho\alpha^*P^* + \beta^*Q^*$.

On the other hand, the optimality of (P^*, Q^*) gives

$$\rho\alpha^*P^* + \beta^*Q^* \leq \rho\alpha^*P' + \beta^*Q',$$

which leads to a contradiction. Thus, (P^*, Q^*) is a Pareto-optimal solution.

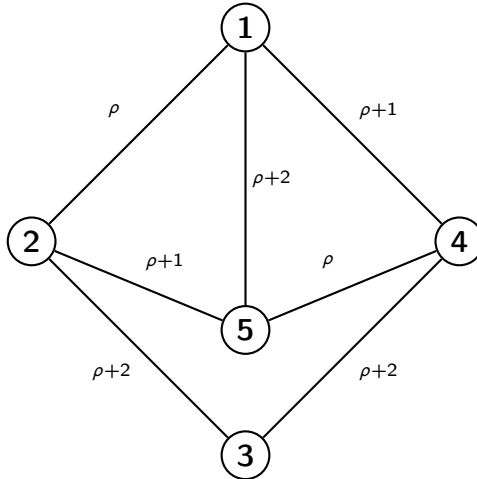
According to Theorem 2, each ρ -NF solution is necessarily a Pareto-optimal solution. □

The set of all Pareto-optimal solutions for BOM is known as the non-dominated set or Pareto set which is here denoted by $S_N \subseteq S$. Let S_ρ be the set of ρ -NF solutions for BOM. In the following, we show that S_ρ is a subset of S_N and this inclusion can be strict.

Theorem 5 For all $\rho > 0$, $S_\rho \subseteq S_N$ and this inclusion can be strict.

Proof According to Proposition 4, we have $S_\rho \subseteq S_N$.

For showing that the set of ρ -NF solutions can be a strict subset of the Pareto set, i.e. $S_\rho \subset S_N$, let us consider an example of the Bi-Objective Travelling Salesman Problem (BOTSP) where we look for a tour minimizing simultaneously the total edge cost and the max-min distance which is the difference between the maximum edge cost and the minimum one [24]. Let G_0 be an undirected graph with each edge cost as follows.



Let (P, Q) denote the solution for the total cost and the max-min distance corresponding to a solution tour. We see that there are two different Hamiltonian tours $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 1$ and $1 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 5 \rightarrow 1$ in G_0 corresponding respectively to two feasible solutions $(P', Q') = (5\rho + 6, 2)$ and $(P'', Q'') = (5\rho + 8, 1)$. Since $P' < P''$ and $Q' > Q''$, (P', Q') and (P'', Q'') are non-dominated solutions with respect to each other. Thus, they are two Pareto-optimal solutions. We have

$$\rho \frac{P''}{P'} + \frac{Q''}{Q'} = \rho \frac{5\rho + 8}{5\rho + 6} + \frac{1}{2} = \rho + 1 + \frac{2\rho}{5\rho + 6} - \frac{1}{2} = \rho + 1 - \frac{\rho + 6}{2(5\rho + 6)} < \rho + 1,$$

Hence, (P', Q') is a Pareto-optimal solution but not a ρ -NF solution.

By contrast, (P'', Q'') is a ρ -NF solution because

$$\rho \frac{P''}{P'} + \frac{Q''}{Q'} = \rho \frac{5\rho+6}{5\rho+8} + \frac{2}{1} = \rho+1+1 - \frac{2\rho}{5\rho+8} = \rho+1 + \frac{3\rho+8}{5\rho+8} > \rho+1,$$

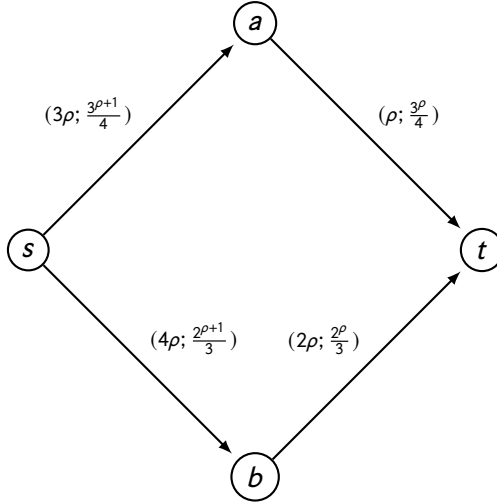
There exist some particular cases where $S_\rho = S_N$. For example, if there is only one Pareto-optimal solution in S_N , it is also a ρ -NF solution. \square

As shown by Theorem 1, there always exists one ρ -NF solution for BOM that minimizes $P^{k_1} Q^{k_2}$ where $k_1 = \frac{\rho}{\rho+1}$ and $k_2 = \frac{1}{\rho+1}$. In the following, we assert that there may be more than one ρ -NF solution for BOM.

Remark 1 *There may be more than one ρ -NF solution for BOM.*

Proof For proving this conclusion, we consider an example of the Bi-Objective Shortest Path Problem (BOSPP) [33]. We first restate the definition of the BOSPP. Let $G = (V, A)$ be a directed graph where V is the set of nodes ($|V| = n$) and A is the set of arcs ($|A| = m$). Let $c_{ij}, t_{ij} > 0$ denote respectively the cost and the time associated with each arc (i, j) . The BOSPP consists in finding a path H from a source node s to a target node t minimizing both criteria $\{ \sum_{(i,j) \in H} c_{ij}, \sum_{(i,j) \in H} t_{ij} \}$.

Now let G_0 be a directed graph with a source node s and a target node t as follows.



Note that each arc of G_0 corresponds respectively to a value of cost and a value of time. For example, the cost associated with arc (s, a) is 3ρ and the time associated with this arc is $\frac{3^{\rho+1}}{4}$. Let (P, Q) denote the solution for the total cost and the total time corresponding to a path from s to t . We see that there are two different paths $s \rightarrow a \rightarrow t$ and $s \rightarrow b \rightarrow t$ in G_0 corresponding respectively to two feasible solutions $(P', Q') = (4\rho, 3^\rho)$ and $(P'', Q'') = (6\rho, 2^\rho)$. Using the Young's inequality for products (see Appendix 1), we have

$$\rho \frac{P''}{P'} + \frac{Q''}{Q'} = \rho \frac{6\rho}{4\rho} + \frac{2^\rho}{3^\rho} = (\rho+1) \left(\frac{\rho}{\rho+1} \times \frac{3}{2} + \frac{1}{\rho+1} \times \left(\frac{2}{3}\right)^\rho \right) \geq (\rho+1) \left(\frac{3}{2}\right)^{\frac{\rho}{\rho+1}} \left(\frac{2}{3}\right)^{\frac{\rho}{\rho+1}} = \rho+1,$$

Thus, (P', Q') is a ρ -NF solution. Moreover, we also have

$$\rho \frac{P'}{P''} + \frac{Q'}{Q''} = \rho \frac{4\rho}{6\rho} + \frac{3\rho}{2\rho} = (\rho+1) \left(\frac{\rho}{\rho+1} \times \frac{2}{3} + \frac{1}{\rho+1} \times \left(\frac{3}{2}\right)^\rho \right) \geq (\rho+1) \left(\frac{2}{3}\right)^{\frac{\rho}{\rho+1}} \left(\frac{3}{2}\right)^{\frac{\rho}{\rho+1}} = \rho+1,$$

Thus, (P'', Q'') is also a ρ -NF solution. Hence, there are two ρ -NF solutions for this example of BOSPP. \square

As shown by Remark 1, there are possibly many ρ -NF solutions for BOM. Among these solutions, two solutions may naturally be preferred by the CDM: the one with the smallest value of P and the other with the smallest value of Q . Let us refer to these solutions as the extreme ρ -NF solutions. In the next section, we present two iterative algorithms for finding extreme ρ -NF solutions in a finite number of iterations.

3 | ALGORITHMS FOR FINDING EXTREME ρ -NF SOLUTIONS

Let us denote P -extreme ρ -NF solution and Q -extreme ρ -NF solution are respectively the ρ -NF solution having the smallest values of P and Q . Notice that they are unique with respect to the values of P and Q since they are both Pareto-optimal solutions. In the following, we will focus first on the P -extreme ρ -NF solution. All the subsequent results applied to the P -extreme ρ -NF solution can be also applied to the Q -extreme ρ -NF solution with slight changes.

Note that if the P -extreme ρ -NF solution and the Q -extreme ρ -NF solution are identical, we obtain a ρ -NF solution having simultaneously the smallest value of P and the smallest value of Q . Thus, it is necessarily a unique ρ -NF solution.

We propose an algorithm to find the P -extreme ρ -NF solution. This iterative algorithm is inspired by the application of the Newton-Raphson method to linear fractional programs that was first discussed by Isbell and Marlow [18] and then generalized to nonlinear fractional programs by Dinkelbach [9]. Starting from $(\alpha_0, \beta_0) = (1, 0)$, we determine the initial solution (P_0, Q_0) which is an optimal solution of $\mathcal{P}_\rho(\alpha_0, \beta_0)$, i.e. the problem minimizing P . Notice that (P_0, Q_0) is not necessarily a Pareto-optimal solution due to $(\alpha_0, \beta_0) = (1, 0) \notin C$. This algorithm can be stated as follows.

Algorithm 1

Input: An instance of BOM, $\rho > 0$, the initial solution (P_0, Q_0) and the corresponding decision vector X_0 .

Output: A decision vector corresponding to the P -extreme ρ -NF solution.

- 1: $i \leftarrow 0$
 - 2: **repeat**
 - 3: $\alpha_{i+1} \leftarrow \frac{Q_i}{P_i+Q_i}, \beta_{i+1} \leftarrow \frac{P_i}{P_i+Q_i}$
 - 4: solve $\mathcal{P}_\rho(\alpha_{i+1}, \beta_{i+1})$ to obtain an optimal solution (P_{i+1}, Q_{i+1}) and the corresponding decision vector X_{i+1}
 - 5: $T_i \leftarrow \rho\alpha_{i+1}P_{i+1} + \beta_{i+1}Q_{i+1} - \rho\alpha_{i+1}P_i - \beta_{i+1}Q_i$
 - 6: $i \leftarrow i + 1$
 - 7: **until** $T_i = 0$
 - 8: **return** $\alpha_i, (P_i, Q_i)$ and X_i .
-

Let $\{\alpha_i\}_{i \geq 0}$ (including $\alpha_0 = 1$) be the sequence of $\{\alpha\}$ constructed by Algorithm 1 and $T_i := \rho\alpha_{i+1}P_{i+1} + \beta_{i+1}Q_{i+1} - \rho\alpha_{i+1}P_i - \beta_{i+1}Q_i$ where $(P_i, Q_i), \forall i \geq 0$ is the optimal solution of $\mathcal{P}_\rho(\alpha_i, \beta_i)$ in the execution of Algorithm 1.

We will prove that Algorithm 1 terminates in a finite number of iterations and the obtained solution is exactly the P -extreme ρ - NF solution. Our proof will use the following lemmas and theorems.

Lemma 6 Let $\alpha', \alpha'' \in [0, 1], \beta' = 1 - \alpha', \beta'' = 1 - \alpha''$ and $(P', Q'), (P'', Q'')$ be respectively the optimal solutions of $\mathcal{P}_\rho(\alpha', \beta')$ and $\mathcal{P}_\rho(\alpha'', \beta'')$. If $\alpha' > \alpha''$ then $P' \leq P''$ and $Q' \geq Q''$.

Proof Since $\alpha', \alpha'' \in [0, 1]$, we also have $\beta', \beta'' \in [0, 1]$.

The optimality of (P', Q') and (P'', Q'') gives

$$\rho\alpha'P' + \beta'Q' \leq \rho\alpha'P'' + \beta'Q'', \text{ and} \quad (4a)$$

$$\rho\alpha''P'' + \beta''Q'' \leq \rho\alpha'P' + \beta''Q' \quad (4b)$$

Multiplying (4a) and (4b) by respectively $\beta'', \beta' \geq 0$ gives

$$\rho\alpha'\beta''P' + \beta'\beta''Q' \leq \rho\alpha'\beta''P'' + \beta'\beta''Q'', \text{ and} \quad (5a)$$

$$\rho\alpha''\beta'P'' + \beta'\beta''Q'' \leq \rho\alpha''\beta'P' + \beta'\beta''Q' \quad (5b)$$

By adding (5a) and (5b), we obtain $\rho(\alpha'\beta'' - \alpha''\beta')(P' - P'') \leq 0$. Since $\alpha' > \alpha''$, we have $\beta'' > \beta' \geq 0$. Thus, $\alpha'\beta'' > \alpha''\beta'$ and it follows that $P' \leq P''$.

In addition, the inequality (4b) implies $\beta''(Q' - Q'') \geq \rho\alpha''(P'' - P') \geq 0$. Since $\beta'' > 0$, we get $Q' \geq Q''$. \square

Lemma 7 During the execution of Algorithm 1, $(\alpha_{i+1}, \beta_{i+1}) \in C, T_i \leq 0$ and the sequence $\{\alpha_i\}$ is strictly decreasing, $\forall i \geq 0$.

Proof Since $P_i, Q_i > 0, \forall i \geq 0$, we have $\alpha_{i+1}, \beta_{i+1} > 0$ and $\alpha_{i+1} + \beta_{i+1} = 1$. Thus, $(\alpha_{i+1}, \beta_{i+1}) \in C, \forall i \geq 0$. It follows that (P_{i+1}, Q_{i+1}) is a Pareto-optimal solution, $\forall i \geq 0$ due to the proof of Proposition 4.

The optimality of (P_{i+1}, Q_{i+1}) gives

$$\rho\alpha_{i+1}P_{i+1} + \beta_{i+1}Q_{i+1} \leq \rho\alpha_{i+1}P_i + \beta_{i+1}Q_i, \forall i \geq 0,$$

Thus, $T_i := \rho\alpha_{i+1}P_{i+1} + \beta_{i+1}Q_{i+1} - \rho\alpha_{i+1}P_i - \beta_{i+1}Q_i \leq 0, \forall i \geq 0$.

We will prove $\alpha_i - \alpha_{i+1} > 0$ by induction on i . If $i = 0$, then $\alpha_0 - \alpha_1 = 1 - \frac{Q_0}{P_0 + Q_0} = \frac{P_0}{P_0 + Q_0} > 0$.

Suppose that our hypothesis is true until $i = k \geq 0$. Since $\alpha_k > \alpha_{k+1}$, we have $P_{k+1} \geq P_k > 0$ and $Q_k \geq Q_{k+1} > 0$ due to Lemma 6. It leads to $Q_k P_{k+1} \geq P_k Q_{k+1}$ and $Q_k P_{k+1} = P_k Q_{k+1} \iff P_k = P_{k+1}$ and $Q_k = Q_{k+1}$, i.e. $(P_k, Q_k) \equiv (P_{k+1}, Q_{k+1})$.

If $T_k = 0$ then Algorithm 1 returns the value α_k . Hence, the sequence $\{\alpha_i\}$ for $i = 0, 1, \dots, k$ is strictly decreasing.

If $T_k < 0$ then we have $(P_k, Q_k) \neq (P_{k+1}, Q_{k+1})$. It implies $Q_k P_{k+1} > P_k Q_{k+1}$. We get

$$\alpha_{k+1} = \frac{Q_k}{P_k + Q_k} > \frac{Q_{k+1}}{P_{k+1} + Q_{k+1}} = \alpha_{k+2}.$$

Thus, in this case our hypothesis is also true with $i = k + 1$. Consequently, $\{\alpha_i\}$ is strictly decreasing, $\forall i \geq 0$. \square

Lemma 8 Algorithm 1 terminates in a finite number of iterations. More precisely, it terminates in at most K iterations where $K \in \mathbb{N}_+ \setminus \{0\}$ represents the number of Pareto-optimal solutions.

Proof Since $\alpha_i > \alpha_{i+1}$, we have $Q_i \geq Q_{i+1}$ and $P_i \leq P_{i+1}, \forall i \geq 0$ due to Lemma 6.

We first show that if $T_i < 0$ then $Q_i > Q_{i+1}, \forall i \geq 0$.

Let us assume that $Q_i = Q_{i+1}$. The optimality of (P_{i+1}, Q_{i+1}) gives

$$\rho\alpha_{i+1}P_{i+1} + \beta_{i+1}Q_{i+1} \leq \rho\alpha_{i+1}P_i + \beta_{i+1}Q_i,$$

Using $Q_i = Q_{i+1}$ and $\rho\alpha_{i+1} > 0$, we obtain $P_{i+1} \leq P_i$. Thus, $P_i = P_{i+1}$.

Since $P_i = P_{i+1}$ and $Q_i = Q_{i+1}$, it implies $T_i = 0$ which leads to a contradiction.

Hence, if $T_i < 0$ then $Q_i > Q_{i+1}$. Consequently, while $T_k < 0$ for some $k \geq 0$, it implies $T_i < 0, \forall 0 \leq i \leq k$ and the execution of Algorithm 1 explores the different Pareto-optimal solutions (P_{i+1}, Q_{i+1}) with the strictly decreasing values of $Q, \forall 0 \leq i \leq k$.

As a result, we obtain $T = 0$ after at most K iterations where $K \in \mathbb{N}_+ \setminus \{0\}$ represents the number of Pareto-optimal solutions. In addition, since \mathcal{S} is a finite set, the Pareto set \mathcal{S}_N is also finite and $K < +\infty$. Thus, Algorithm 1 terminates in a finite number of iterations. \square

Theorem 9 *The solution given by Algorithm 1 is the P -extreme ρ -NF solution.*

Proof Let (P_n, Q_n) be the solution given by Algorithm 1. Since $T_n = 0$, (P_n, Q_n) is a ρ -NF solution due to Proposition 3.

If $n = 0$ then (P_0, Q_0) is obviously the P -extreme ρ -NF solution as it is an optimal solution of the problem minimizing P . We consider $n \geq 1$ that implies $T_n = 0$ and $T_i < 0, \forall 0 \leq i \leq n - 1$.

We will prove that (P_n, Q_n) is the ρ -NF solution with the smallest value of P . Suppose that it is not true. Thus, there exists a ρ -NF solution (P^*, Q^*) such that $P^* < P_n$. According to Theorem 2, there exists $(\alpha^*, \beta^*) \in C$ such that (P^*, Q^*) is an optimal solution of $\mathcal{P}_\rho(\alpha^*, \beta^*)$ and $\alpha^*P^* = \beta^*Q^*$. Since $P^* < P_n$, we have $\alpha^* \geq \alpha_n$ due to Lemma 6. Since the sequence $\{\alpha\}$ is strictly decreasing and $\alpha^* < \alpha_0 = 1$, there exists $0 \leq i \leq n - 1$ such that $\alpha^* \in [\alpha_{i+1}, \alpha_i]$.

We first show that $\alpha^* \neq \alpha_{i+1}$ by contradiction. Let assume that $\alpha^* = \alpha_{i+1}$. We also have $\beta^* = \beta_{i+1}$ that leads to $\alpha_{i+1}P^* = \beta_{i+1}Q^*$. Thus, we get

$$\frac{P^*}{Q^*} = \frac{\beta_{i+1}}{\alpha_{i+1}} = \frac{P_i}{Q_i} \implies \frac{P^*}{P_i} = \frac{Q^*}{Q_i},$$

If $i = 0$ then $\frac{P^*}{P_0} = \frac{Q^*}{Q_0}$. Since $P^* \geq P_0$, we get $Q^* \geq Q_0$. Thus, $\rho\alpha_1P^* + \beta_1Q^* \geq \rho\alpha_1P_0 + \beta_1Q_0$.

If $i \geq 1$ then $P^* = P_i, Q^* = Q_i$ since they are both Pareto-optimal solutions and $\frac{P^*}{P_i} = \frac{Q^*}{Q_i}$. Thus, $\rho\alpha_{i+1}P^* + \beta_{i+1}Q^* = \rho\alpha_{i+1}P_i + \beta_{i+1}Q_i$.

Consequently, we always have $\rho\alpha_{i+1}P^* + \beta_{i+1}Q^* \geq \rho\alpha_{i+1}P_i + \beta_{i+1}Q_i$. Moreover, $\rho\alpha_{i+1}P_{i+1} + \beta_{i+1}Q_{i+1} = \rho\alpha_{i+1}P^* + \beta_{i+1}Q^*$ because both (P_{i+1}, Q_{i+1}) and (P^*, Q^*) are the optimal solutions of $\mathcal{P}_\rho(\alpha_{i+1}, \beta_{i+1})$. Thus, $T_i = \rho\alpha_{i+1}P_{i+1} + \beta_{i+1}Q_{i+1} - \rho\alpha_{i+1}P_i - \beta_{i+1}Q_i \geq 0$ which leads to a contradiction due to $T_i < 0, \forall 0 \leq i \leq n - 1$. Hence, we have $\alpha^* \neq \alpha_{i+1}$.

It follows that $\alpha^* \in (\alpha_{i+1}, \alpha_i)$. Since $\alpha^* < \alpha_i$, we have $P^* \geq P_i$ and $Q^* \leq Q_i$ due to Lemma 6. Thus, we get

$$\alpha^* = \frac{Q^*}{P^* + Q^*} \leq \frac{Q_i}{P_i + Q_i} = \alpha_{i+1},$$

which leads to a contradiction due to the fact that $\alpha^* > \alpha_{i+1}$. Hence, (P_n, Q_n) is a ρ -NF solution having the smallest value of P , i.e. the P -extreme ρ -NF solution. \square

For finding the Q -extreme ρ -NF solution, we use the similar algorithm starting from $(\alpha_0, \beta_0) = (0, 1)$ instead of $(1, 0)$ and (P_0, Q_0) as an optimal solution of $\mathcal{P}_\rho(\alpha_0, \beta_0)$, i.e. the problem minimizing Q . It can be stated as follows.

Algorithm 2

Input: An instance of BOM, $\rho > 0$, the initial solution (P_0, Q_0) and the corresponding decision vector X_0 .

Output: A decision vector corresponding to the Q -extreme ρ -NF solution.

```

1:  $i \leftarrow 0$ 
2: repeat
3:    $\alpha_{i+1} \leftarrow \frac{Q_i}{P_i+Q_i}, \beta_{i+1} \leftarrow \frac{P_i}{P_i+Q_i}$ 
4:   solve  $\mathcal{P}_\rho(\alpha_{i+1}, \beta_{i+1})$  to obtain an optimal solution  $(P_{i+1}, Q_{i+1})$  and the corresponding decision vector  $X_{i+1}$ 
5:    $T_i \leftarrow \rho\alpha_{i+1}P_{i+1} + \beta_{i+1}Q_{i+1} - \rho\alpha_{i+1}P_i - \beta_{i+1}Q_i$ 
6:    $i \leftarrow i + 1$ 
7: until  $T_i = 0$ 
8: return  $\alpha_i, (P_i, Q_i)$  and  $X_i$ .
```

We also state some lemmas and theorems to prove that we obtain the Q -extreme ρ -NF solution by using Algorithm 2.

Lemma 10 *During the execution of Algorithm 2, $(\alpha_{i+1}, \beta_{i+1}) \in C$, $T_i \leq 0$ and the sequence $\{\alpha_i\}$ is strictly increasing, $\forall i \geq 0$.*

Lemma 11 *Algorithm 2 terminates in a finite number of iterations. More precisely, it terminates in at most K iterations where $K \in \mathbb{N}_+ \setminus \{0\}$ represents the number of Pareto-optimal solutions.*

Theorem 12 *The solution given by Algorithm 2 is the Q -extreme ρ -NF solution.*

4 | EFFECT OF ρ ON THE VALUES OF EXTREME ρ -NF SOLUTIONS

In this section, we show the effect of ρ on the values of the corresponding P -extreme ρ -NF solution and Q -extreme ρ -NF solution. In fact, when ρ becomes larger, the CDM prefers an alternative assigning a smaller value of P and the value of P in the P -extreme ρ -NF solution becomes closer to the value of P in the optimal solution of the problem minimizing P . Furthermore, when ρ is large enough, they are identical. Conversely, when ρ becomes smaller, an alternative assigning a smaller value of Q is preferred and the value of Q in the Q -extreme ρ -NF solution becomes closer to the value of Q in the optimal solution of the problem minimizing Q . When ρ is small enough, they are also identical. Hence, this helps the CDM to choose an appropriate parameter ρ corresponding to her own preferred solutions based on the values of P and Q . In addition, For proving this conclusion, we state the following lemma and theorems.

Lemma 13 *Given $\rho', \rho'' > 0$. Let $\alpha', \alpha'' \in [0, 1]$, $\beta' = 1 - \alpha', \beta'' = 1 - \alpha''$ and $(P', Q'), (P'', Q'')$ be respectively the optimal solutions of $\mathcal{P}_{\rho'}(\alpha', \beta')$ and $\mathcal{P}_{\rho''}(\alpha'', \beta'')$. If $\rho' \alpha' \beta'' > \rho'' \alpha'' \beta'$ then $P' \leq P''$ and $Q' \geq Q''$.*

Proof In fact, this Lemma is a generalized version of Lemma 6 where we consider two different values of ρ . Using the same procedures as Lemma 6, we obtain $(\rho' \alpha' \beta'' - \rho'' \alpha'' \beta')(P' - P'') \leq 0$. Since $\rho' \alpha' \beta'' > \rho'' \alpha'' \beta'$, it follows that $P' \leq P''$.

In addition, we also have $\beta''(Q' - Q'') \geq \rho''\alpha''(P'' - P') \geq 0$. Since $\rho'\alpha'\beta'' > \rho''\alpha''\beta' \geq 0$, we get $\beta'' > 0$. Hence, $Q' \geq Q''$. \square

Theorem 14 Given $\rho' > \rho'' > 0$ and let $(P', Q'), (P'', Q'') \in S$ be respectively the P -extreme ρ' -NF solution and the P -extreme ρ'' -NF solution. Then $P' \leq P''$.

Proof Suppose that Algorithm 1 returns (P', Q') and (P'', Q'') in respectively n' and n'' iterations where $n', n'' \in \mathbb{N}_+ \setminus \{0\}$. Denote $\{\alpha'_i\}_{0 \leq i \leq n'}$ and $\{\alpha''_i\}_{0 \leq i \leq n''}$ as two sequences constructed by Algorithm 1. Without loss of generality, we assume that $n' \geq n''$. We first extend the sequence $\{\alpha''\}$ such that it has the same number of elements as $\{\alpha'\}$. In case $n'' < n'$, we add the element $\alpha''_{n''+1} = \frac{Q''}{P''+Q''}$ to $\{\alpha''\}$ and repeat $\alpha''_{n''+1}$ such that both the sequence $\{\alpha''\}$ and the sequence $\{\alpha'\}$ have $n' + 1$ elements. We also repeat the optimal solution (P'', Q'') corresponding to each α''_i where $i \geq n'' + 1$. Let (P'_i, Q'_i) and (P''_i, Q''_i) be respectively the optimal solutions of $\mathcal{P}_{\rho'}(\alpha'_i, \beta'_i)$ and $\mathcal{P}_{\rho''}(\alpha''_i, \beta''_i)$, $\forall 0 \leq i \leq n' + 1$ as constructed above. Then, $(P'_{n'}, Q'_{n'})$ and $(P''_{n'}, Q''_{n'})$ are respectively the P -extreme ρ' -NF solution and the P -extreme ρ'' -NF solution. (In fact, we have $P'_i = P'', Q'_i = Q'', \forall i \geq n'' + 1$ and $P'_{n'} = P', Q'_{n'} = Q'$)

As both (α'_0, β'_0) and (α''_0, β''_0) are the initial coefficients for using Algorithm 1, we have $\alpha'_0 = \alpha''_0 = 1$ and $\beta'_0 = \beta''_0 = 0$. Note that starting from an arbitrary optimal solution of the problem minimizing P , Algorithm 1 always returns the P -extreme ρ -NF solution. Thus, we can assume that $P'_0 = P''_0$ and $Q'_0 = Q''_0$. We prove $\rho'\alpha'_i\beta''_i > \rho''\alpha''_i\beta'_i, \forall 1 \leq i \leq N$ by induction on i . For $i = 1$, since $P'_0 = P''_0$ and $Q'_0 = Q''_0$, we have $\alpha'_1 = \alpha''_1 > 0$ and $\beta'_1 = \beta''_1 > 0$. Hence, $\rho'\alpha'_1\beta''_1 > \rho''\alpha''_1\beta'_1$. Suppose that our hypothesis is true until $i = k \geq 1$, we will show that it is also true with $i = k + 1$.

The inductive hypothesis gives $\rho'\alpha'_k\beta''_k > \rho''\alpha''_k\beta'_k$ that implies $P'_k \leq P''_k$ and $Q'_k \geq Q''_k$ due to Lemma 13. Thus, we get

$$\alpha'_{k+1} = \frac{Q'_k}{P'_k + Q'_k} \geq \frac{Q''_k}{P''_k + Q''_k} = \alpha''_{k+1}. \quad (6)$$

Since $\alpha'_{k+1} \geq \alpha''_{k+1} > 0$, we obtain $0 < \beta'_{k+1} \leq \beta''_{k+1}$. Consequently, $\rho'\alpha'_{k+1}\beta''_{k+1} > \rho''\alpha''_{k+1}\beta'_{k+1}$ and our hypothesis is true with $i = k + 1$.

Hence, $\rho'\alpha'_n\beta''_n > \rho''\alpha''_n\beta'_n$ and we obtain $P'_n \leq P''_n$ due to Lemma 13. That concludes the proof. \square

The result for the Q -extreme ρ -NF solution can be proved similarly. We state the following theorem for that.

Theorem 15 Given $\rho' > \rho'' > 0$ and let $(P', Q'), (P'', Q'') \in S$ be respectively the Q -extreme ρ' -NF solution and the Q -extreme ρ'' -NF solution. Then $Q'' \leq Q'$.

As a consequence of Theorems 14 and 15, when ρ (or $1/\rho$) becomes larger, the P -extreme ρ -NF solution becomes closer to the Q -extreme ρ -NF solution. That is to say, the ρ -NF solution tends to be unique when ρ is further from 1.

5 | COMPUTATIONAL RESULTS

5.1 | Bi-Objective Travelling Salesman Problem (BOTSP)

5.1.1 | Definition and modeling

Let us first restate the BOTSP introduced in [24]. The BOTSP is a variant of the TSP where we find a tour minimizing simultaneously the total cost and the max-min distance. The latter is the difference between the maximum edge cost and the minimum one in the tour.

Note that the problem minimizing only the max-min distance, called the Balanced TSP (BTSP), has been introduced by Larusic and Punnen [20] for finding Hamiltonian tours in several cases where the equitable distribution of edges are important, for example, the nozzle guide vane assembly problem [26] and the cyclic workforce scheduling problem [37].

Given an undirected graph $G = (V, E)$ where $|V| = n$, $|E| = m$, $c_e \in \mathbb{R}_+ \setminus \{0\}$ is a cost associated with every edge $e \in E$ and $\Pi(G) \neq \emptyset$ denote the set of all Hamiltonian cycles in G . The BOTSP can be formally formulated as

$$\min_{H \in \Pi(G)} P = \sum_{e \in H} c_e \quad (7a)$$

$$\min_{H \in \Pi(G)} Q = \max_{e \in H} c_e - \min_{e' \in H} c_{e'} \quad (7b)$$

For solving $\mathcal{P}_\rho(\alpha, \beta)$, we present a MIP formulation as follows:

$$\min \rho\alpha P + \beta Q \quad (8a)$$

$$\text{s.t. } \sum_{e \in \delta(v)} x_e = 2 \quad \forall v \in V \quad (8b)$$

$$\sum_{e \in \delta(V')} x_e \geq 2 \quad \forall \emptyset \neq V' \subseteq V \quad (8c)$$

$$Q \geq u - l \quad (8d)$$

$$u \geq c_e x_e \quad \forall e \in E \quad (8e)$$

$$l \leq c_e x_e + (1 - x_e)M \quad \forall e \in E \quad (8f)$$

$$x_e \in \{0, 1\} \quad \forall e \in E \quad (8g)$$

where x_e is the binary variable representing the occurrence of e in the solution tour, $\delta(v)$ is the incident edges set of vertex v , $\delta(V')$ is the set of edges which have exactly one endpoint in V' , M is the largest edge cost (i.e. $M = \max_{e \in E} c_e$). Constraints (8b) are respectively the degree constraints which assure that there are exactly two edges incident to every vertex. Constraints (8c) are the subtour elimination constraints. These constraints represent the classical Held-Karp (linear programming) relaxation for TSP. Together with Constraints (8g), they assure that the solution is a tour. In order to calculate the max-min distance Q , we need to determine the maximum and the minimum edge costs u and l in the solution tour. Constraints (8e) obviously allow to bound u from below by the largest weight edge in the solution tour. Constraints (8f) allow to bound l from above by the minimum edge cost in the solution tour. Indeed, in the case where $x_e = 1$, Constraints (8f) guarantee that l is smaller than the costs of all edges in the tour. Otherwise, when x_e equals 0, the largest edge cost M assures the validity of Constraints (8f). As $\rho\alpha P + \beta Q$ is minimized, u and l will respectively take the values of the maximum and the minimum edge costs.

5.1.2 | Computational results on the instances of BOTSP

In this section, we present computational results on some instances of the above BOTSP. Let us denote NFTSP as the problem of finding extreme ρ -NF solutions for the BOTSP. We conduct several experiments aiming at solving the NFTSP with Algorithms 1 and 2 on rather small size instances from the TSPLIB [30]. We also solve the TSP and the BTSP on the same instances. Note that both the TSP and the Balanced TSP can be solved as special cases of $\mathcal{P}_\rho(\alpha, \beta)$ with $(\alpha, \beta) = (1, 0)$ and $(0, 1)$, respectively. The obtained solutions for three problems will be then compared and

commented.

For solving three problems TSP, BTSP and NFTSP, we design a simple branch-and-cut algorithm devoted to minimize a linear objective function over the MIP formulation. Note that Constraints (8e) and (8f) are excluded for the TSP. We use CPLEX 12.10 to implement our branch-and-cut algorithm. Constraints (8c) are set as lazy cuts which are generated only when being violated by some integer solution. For the BTSP and the NFTSP, we also have some specific branching rules for variable l inspired from some threshold algorithms [20], [21]. For the NFTSP, this branch-and-cut algorithm is used in each iteration of Algorithms 1 and 2 to solve the subproblem $\mathcal{P}_\rho(\alpha, \beta)$. All the experiments are conducted on a PC Intel Core i5-9500 3.00GHz with 6 cores and 6 threads.

TABLE 1 Optimal solutions for TSP and BTSP

Instance	TSP			BTSP		
	P	Q	Time	P	Q	Time
burma14	3323	472	0.06	4986	134	0.03
ulysses16	6859	1452	0.14	13988	868	0.05
gr17	2085	311	0.04	4310	119	0.07
gr21	2707	328	0.01	8645	115	0.10
ulysses22	7013	1490	10.17	19001	868	0.09
gr24	1272	83	0.01	3926	33	0.18
fri26	937	118	0.02	2460	21	0.24
bays29	2020	140	0.04	6810	38	0.62
bayg29	1610	86	0.03	4210	29	0.59

TABLE 2 P -extreme ρ -NF solutions for BOTSP

Instance	$\rho = 1$				$\rho = \log_2 n$				$\rho = 1/\log_2 n$			
	P	Q	Time	Iters	P	Q	Time	Iters	P	Q	Time	Iters
burma14	4986	134	0.17	4	3558	294	0.11	3	4986	134	0.36	3
ulysses16	7047	1399	0.80	3	6859	1452	0.37	2	13670	868	0.72	3
gr17	2227	234	1.43	3	2090	262	0.20	3	4029	119	1.01	4
gr21	2989	278	0.63	3	2709	326	0.05	3	5945	120	0.76	3
ulysses22	7070	1471	67.31	3	7013	1490	26.79	2	18613	868	3.64	4
gr24	1282	81	0.73	3	1272	83	0.10	2	3847	33	11.85	4
fri26	980	82	1.29	3	953	91	0.26	3	2447	21	14.74	4
bays29	3449	59	50.52	5	2020	140	0.24	2	5384	40	75.78	4
bayg29	1817	63	5.40	3	1610	86	0.26	2	4210	29	37.26	4

We assume that the CDM considers three different values of ρ , respectively $1, \log_2 n$ and $1/\log_2 n$ where n is the number of nodes in graph G . Notice that the selection for the values of ρ depends on the CDM's own strategy and it does not affect the Pareto set. In case $\rho = 1$, the two objectives P and Q are equivalently important. In case $\rho = \log_2 n$ (or $\rho = 1/\log_2 n$), the CDM gives priority to the value of P (or the value of Q) by the relative importance $\log_2 n$. We show the optimal solutions for the TSP and the BTSP in Table 1. Table 2 and Table 3 present respectively the P -extreme ρ -NF solutions and the Q -extreme ρ -NF solutions for NFTSP in several instances from the TSPLIB with a range of nodes from 14 to 29. We provide the number of iterations for the NFTSP in the columns "Iters".

We can see by the values of P and Q in these tables that the P -extreme ρ -NF and Q -extreme ρ -NF solutions for the BOTSP normally strike a better trade-off between two objectives: the total cost and the max-min distance comparing with those in the optimal solutions for the TSP and the BTSP. More precisely, when $\rho = 1$, the P -extreme

TABLE 3 Q -extreme ρ - NF solutions for BOTSP

Instance	$\rho = 1$				$\rho = \log_2 n$				$\rho = 1/\log_2 n$			
	P	Q	Time	Iters	P	Q	Time	Iters	P	Q	Time	Iters
burma14	4986	134	0.06	2	4901	142	0.19	3	4986	134	0.33	2
ulysses16	13670	868	0.95	2	6859	1452	0.83	4	13670	868	0.06	2
gr17	3346	139	0.30	3	2090	262	0.61	4	4029	119	0.73	3
gr21	5945	120	0.46	3	2709	326	1.20	4	5945	120	0.42	3
ulysses22	7070	1471	74.37	4	7013	1490	51.37	4	18613	868	0.10	3
gr24	3847	33	2.42	2	1272	83	0.44	4	3847	33	0.04	2
fri26	2447	21	0.25	2	953	91	8.37	5	2447	21	0.10	3
bays29	4558	44	10.05	3	2093	116	29.92	5	6714	38	0.93	3
bayg29	3246	35	20.84	4	1610	86	3.20	5	4210	29	0.36	2

ρ - NF solutions (respectively the Q -extreme ρ - NF solutions) offer a better alternative than the optimal solutions of the TSP (respectively the BTSP) with a significant drop on the values of Q (respectively P) and a slight growth on the values of P (respectively Q). Table 2 and Table 3 also indicate that Algorithms 1 and 2 seem to converge quickly after only maximum 5 iterations. In general, they terminate in a polynomial number of iterations. For proving this, we state the following lemma. Note that the result for Algorithm 2 can be proved similarly.

Lemma 16 For the BOTSP, Algorithm 1 returns the P -extreme ρ - NF solution in a polynomial number of iterations.

Proof We first show that the number of Pareto-optimal solutions for the BOTSP is at most $C_{C_n}^2 = \frac{1}{8}n(n^2 - 1)(n + 2)$ where n is the number of nodes. Let c_i^{max} and c_i^{min} be the maximum and the minimum edge cost in the tour corresponding to the Pareto-optimal solution (P_i, Q_i) then $Q_i = c_i^{max} - c_i^{min}$. For two distinct Pareto-optimal solutions $(P_i, Q_i), (P_j, Q_j)$, we have $Q_i \neq Q_j$ which is equivalent to $c_i^{max} - c_i^{min} \neq c_j^{max} - c_j^{min}$. Thus, $(c_i^{max}, c_i^{min}) \neq (c_j^{max}, c_j^{min})$. Consequently, each Pareto-optimal solution obtained by an iteration of Algorithm 1 has distinct pair of edges corresponding to the maximum edge cost and the minimum one.

Note that for the graph G with n nodes, we have at most C_n^2 edges and then the distinct pairs of edges is at most $C_{C_n}^2$. Hence, the number of Pareto-optimal solutions for the BOTSP is at most $C_{C_n}^2 = \frac{1}{8}n(n^2 - 1)(n + 2)$ which implies that Algorithm 1 terminates in a polynomial number of iterations due to the proof of Lemma 8. \square

When $\rho = \log_2 n$, the values of P in the obtained P -extreme ρ - NF are closer to those in the optimal solutions for the TSP comparing to $\rho = 1$ and $\rho = 1/\log_2$. Moreover, we need less (or more) iterations for finding the P -extreme ρ - NF solutions when ρ becomes larger (or smaller). Similarly, when $\rho = 1/\log_2 n$, the values of Q in the obtained Q -extreme ρ - NF are closer to those in the optimal solutions for the BTSP comparing to $\rho = \log_2 n$ and $\rho = 1$. We also need less (or more) iterations for finding the Q -extreme ρ - NF solutions when ρ becomes smaller (or larger).

In addition, when $\rho = \log_2 n$ and $\rho = 1/\log_2 n$, the P -extreme ρ - NF solutions and the Q -extreme ρ - NF solutions are identical in most of instances. Thus, for such instances we only have a unique ρ - NF solution. In general, when ρ (or $1/\rho$) becomes larger, the P -extreme ρ - NF solution is closer to the Q -extreme ρ - NF solution. That is to say, ρ - NF solution tends to be unique when ρ becomes further from 1. Note that the CPU time for solving the NFTSP in some instances is large comparing with the CPU time spent for solving the TSP and the BTSP. A deeper analysis on the iterations of Algorithms 1 and 2 tells us that the smaller is the value of α , the more time is needed for solving $\mathcal{P}_\rho(\alpha, \beta)$.

5.2 | Bi-Objective Shortest Path Problem (BOSPP)

5.2.1 | Definition and modeling

We first restate the BOSPP that we have utilized in Section 2.2. The BOSPP is a variant of the shortest path problem which is cited as the most common problem in combinatorial optimization [4] due to its numerous applications as the minimum cost path problem, the quickest path problem, the most reliable path problem and various routing problems. As one of simplest problems in multicriterion linear integer programming, the BOSPP refers to find the shortest path between two nodes satisfying two objectives, such as "cheap and quickest", "cheap and more reliable", "cheap and energy efficient", and so on. More precisely, we describe the BOSPP as follows.

Let $G = (V, A)$ be a directed graph where V is the set of nodes ($|V| = n$) and A is the set of arcs ($|A| = m$). Let $c_{ij}, t_{ij} > 0$ denote respectively the cost and the time associated with each arc (i, j) . The BOSPP consists in finding a path H from a source node s to a target node t minimizing both criteria $\{\sum_{(i,j) \in H} c_{ij}, \sum_{(i,j) \in H} t_{ij}\}$. Notice that although there exist efficient exact algorithms for solving the single-objective shortest path problem, the multi-objective case (including the bi-objective case) is known to be NP-complete by transformation from a 0-1 knapsack problem [12], [16].

Let P and Q be the objective functions for the total cost and the total time corresponding to the s - t path. As shown in Algorithm 1, for finding the extreme ρ -NF solutions for the BOSPP, we need to solve $\mathcal{P}_\rho(\alpha, \beta)$ as a sequence of convex combinations of ρP and Q , i.e. including $(\alpha, \beta) = (1, 0), (0, 1)$. For each value of (α, β) , we have $\rho\alpha P + \beta Q = \sum_{(i,j) \in A} (\rho\alpha c_{ij} + \beta t_{ij})x_{ij}$. Thus, we construct a graph G' with the same sets of nodes and arcs of G , i.e. $G' = (V, A)$, and $\rho\alpha c_{ij} + \beta t_{ij}$ is a unique value of cost associated with each arc (i, j) of G' . Hence, solving $\mathcal{P}_\rho(\alpha, \beta)$ is now equivalent to solve the single-objective shortest path problem in G' .

5.2.2 | Computational results on the instances of BOSPP

Let us denote NFSPP as the problem of finding extreme ρ -NF solutions for the BOSPP. We investigate the performance of the presented algorithms for the NFSPP on random NETMAKER networks [32]. NETMAKER networks generator was first proposed in [32] to test a label-correcting algorithm for the BOSPP. It was also used in [27], [28] to evaluate the performance of algorithms for the BOSPP. To generate a NETMAKER network, a random Hamiltonian cycle is first constructed to assure that the network is strongly connected. Let $\{1, \dots, n\}$ be the node set of the network. For each node i , a random number of i 's outgoing arcs, which belongs to the interval $\{a_{\min}, \dots, a_{\max}\}$, is uniformly generated. Moreover, node i is only allowed to reach nodes j with $j \in \{i - \lfloor \frac{I_{node}}{2} \rfloor, i + \lfloor \frac{I_{node}}{2} \rfloor\}$ where I_{node} is the node interval. The cost and time associated with an arc are chosen randomly in two intervals $\{1, \dots, 33\}$ and $\{67, \dots, 100\}$ such that two values belong to different intervals. Table 4 describes the generated NETMAKER networks. Notice that for all the instances, the source node is 1 and the target node is n . To solve the single-objective shortest path problem, we use the Dijkstra's algorithm [8] implemented in Networkx package version 2.5.1 [14]. All the experiments are conducted on a PC Intel Core i5-9500 3.00GHz with 6 cores and 6 threads.

We assume that the CDM considers three different values of ρ , respectively 1, 1/2 and 2. The optimal solutions for the problem minimizing P and the problem minimizing Q are shown in Table 5. Table 6 and Table 7 present respectively the P -extreme ρ -NF solutions and the Q -extreme ρ -NF solutions. We provide the number of iterations for solving the NFSPP in the columns "Iters".

Similar to the above results for the BOTSP, when $\rho = 1$, the P -extreme ρ -NF solutions (respectively the Q -extreme ρ -NF solutions) offer a better alternative than the optimal solutions of the problem minimizing P (respectively the

TABLE 4 Random NETMAKER networks

Name	Nodes	I_{node}	Outgoing arcs		Arcs
			a_{min}	a_{max}	
N1.1	1000	20	5	15	8319
N1.2	1000	20	1	20	8290
N1.3	1000	50	5	15	9465
N5.1	5000	20	5	15	41438
N5.2	5000	20	1	20	41787
N10.1	10000	20	5	15	82785
N10.2	10000	20	1	20	83222
N20.1	20000	20	5	15	165702
N20.2	20000	20	1	20	166086

TABLE 5 Optimal solutions of the problems minimizing P and Q

Instance	Min P			Min Q		
	P	Q	Time	P	Q	Time
N1.1	1521	14482	0.03	13815	1541	0.03
N1.2	1562	14443	0.03	14057	1516	0.03
N1.3	489	5539	0.03	6067	531	0.03
N5.1	7906	71535	0.17	70702	7902	0.18
N5.2	8385	70901	0.17	72431	7899	0.18
N10.1	16158	145440	0.44	139676	15728	0.42
N10.2	15904	139863	0.43	141761	16330	0.44
N20.1	31482	286335	1.56	287844	31645	1.66
N20.2	32559	285324	1.56	285408	33389	1.72

problem minimizing Q) with a significant drop on the values of Q (respectively P) and a slight growth on the values of P (respectively Q). Generally, the numbers of iterations for Algorithms 1 and 2 are much more than those for the BOTSP. In fact, it may be difficult to determine a good upper bound on the number of iterations for Algorithms 1 and 2 since the values of P and Q in the BOSPP are totally independent (unlike the BOTSP). However, the quick CPU times spent even for some instances with 20000 nodes show the efficiency of our algorithms combining with the Dijkstra's algorithm for solving the NFSPP.

When $\rho = 2$, the values of P in the obtained P -extreme ρ -NF are closer to those in the optimal solutions for the problem minimizing P comparing to $\rho = 1$ and $\rho = 1/2$. In most cases, we need less (or more) iterations for finding the P -extreme ρ -NF solutions when ρ becomes larger (or smaller). Similarly, when $\rho = 1/2$, the values of Q in the obtained Q -extreme ρ -NF are closer to those in the optimal solutions for the problem minimizing Q comparing to $\rho = 2$ and $\rho = 1$. We also need less (or more) iterations for finding the Q -extreme ρ -NF solutions when ρ becomes smaller (or larger).

When $\rho = 2$ and $\rho = 1/2$, the P -extreme ρ -NF solutions and the Q -extreme ρ -NF solutions are identical in most of instances. Thus, there is only a unique ρ -NF solution for such instances. Generally, when ρ (or $1/\rho$) becomes larger, the P -extreme ρ -NF solution is closer to the Q -extreme ρ -NF solution. In other words, ρ -NF solution tends to be unique when ρ becomes further from 1.

TABLE 6 P -extreme ρ -NF solutions for BOSPP

Instance	$\rho = 1$				$\rho = 1/2$				$\rho = 2$			
	P	Q	Time	iter	P	Q	Time	Iters	P	Q	Time	Iters
N1.1	1629	12339	0.12	4	13089	1564	0.40	13	1556	13097	0.10	3
N1.2	1652	12199	0.12	4	2190	10069	0.17	6	1594	12952	0.12	4
N1.3	496	5166	0.11	3	650	4335	0.17	5	496	5166	0.10	3
N5.1	8655	59520	1.10	6	63466	8083	2.27	13	8141	64727	0.90	5
N5.2	9383	58079	1.06	6	65438	8049	1.92	11	8519	65782	0.71	4
N10.1	17625	117018	2.08	5	126963	16072	5.29	13	16681	126573	2.99	7
N10.2	17127	118037	1.66	4	127784	16777	5.55	14	16237	127635	2.13	5
N20.1	34321	234978	7.15	5	253918	32603	17.62	13	32413	253953	7.68	5
N20.2	35186	239017	9.17	6	255794	34265	18.20	13	33307	258202	6.68	4

TABLE 7 Q -extreme ρ -NF solutions for BOSPP

Instance	$\rho = 1$				$\rho = 1/2$				$\rho = 2$			
	P	Q	Time	Iters	P	Q	Time	Iters	P	Q	Time	Iters
N1.1	11778	1688	0.12	4	13089	1564	0.09	3	1556	13097	0.45	15
N1.2	12506	1609	0.13	4	13438	1525	0.10	3	9786	2366	0.21	7
N1.3	4987	565	0.10	3	5093	558	0.10	3	4713	598	0.10	3
N5.1	59944	8407	0.71	4	63466	8083	0.90	5	8151	64568	2.59	15
N5.2	59608	8661	1.05	6	65438	8049	0.71	4	8519	65782	2.25	13
N10.1	118773	16855	1.99	5	126963	16072	1.67	4	16681	126573	5.13	13
N10.2	118563	17777	2.48	6	127784	16777	2.07	5	16237	127635	5.07	13
N20.1	235196	34504	7.20	5	253918	32603	7.41	5	32413	253953	17.71	13
N20.2	235068	36483	13.48	9	255794	34265	7.67	5	33313	258109	16.86	12

6 | CONCLUSION

In this paper, we have utilized proportional Nash equilibrium adapted in context of *Bi-Objective Minimization* (BOM) to propose a novel criterion for selecting the preferred solutions in the Pareto set. For this purpose, we have generalized the concept of *Nash Fairness* (NF) solution introduced in our conference papers, called ρ -NF solution, by adding a parameter $\rho > 0$ which reflects the relative importance of one objective to the other one. We have first proven the existence of ρ -NF solutions for BOM. Especially, the set of ρ -NF solutions can be a strict subset of the Pareto set. Second, we have designed two algorithms to find extreme ρ -NF solutions for BOM. Based on the Weighted Sum Method, these algorithms only require minimizing a sequence of linear combinations of the objectives. In addition, computational results conducted on some instances of the BOTSP and the BOSPP have shown the efficiency of our algorithms as they seem to converge quickly and the obtained extreme ρ -NF solutions have better trade-off between two objectives.

Our future works are determining an efficient upper bound on the number of iterations in our algorithms and showing that the set of ρ -NF solutions is significantly smaller than the Pareto set in general. We also want to generalize the concept of ρ -NF solution for *Bi-Objective Optimization* (BOO) whose objectives can be either maximized or minimized. More precisely, we aim at visualizing the ρ -NF frontier, i.e. the set of ρ -NF solutions, for BOO.

References

- [1] D. Bertsimas, V. F.Farias, and N. Trichakis, *The price of fairness*, Oper. Res. **59-1** (January-February 2011), 17–31.
- [2] A. Charnes, R.W. Clower, and K.O. Kortanek, *Effective control through coherent decentralization with preemptive goals*, Econometrica **35** (1967), 294–320.
- [3] A. Charnes, W.W. Cooper, and R.O. Ferguson, *Optimal estimation of executive compensation by linear programming*, Manage. Sci. **1** (1955), 138–151.
- [4] Y. Chen and Y. Chin, *The quickest path problem*, Comput. Oper. Res. **17** (1990), 153–161.
- [5] J.L. Cohon, *Multi-objective programming and planning*, New York: Academic Press (1983).
- [6] I. Das and J.E. Dennis, *Normal-boundary intersection: A new method for generating the pareto surface in nonlinear multicriteria optimization problems*, SIAM J. Optim. **8-3** (1998), 631.
- [7] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, *A fast and elitist multiobjective genetic algorithm: Nsga-ii*, IEEE Trans. Evolutionary Computation **6-2** (2002), 182.
- [8] E.W. Dijkstra, *A note on two problems in connexion with graphs*, Numer. Mathematik **1** (1959), 269–271.
- [9] W. Dinkelbach, *On nonlinear fractional programming*, Manage. Sci. **13** (1967), 492–498.
- [10] G.S. Ernando, C.S. Andrea, and J.A. Dario, *An exact method for solving the bi-objective minimum diameter-cost spanning tree problem*, RAIRO-Operations Res. **49** (2015), 143–160.
- [11] P.C. Fishburn, *Lexicographic orders, utilities, and decision rules: A survey*, Manage. Sci. (1974), 1442–1471.
- [12] M. Garey and D. Johnson, *Computers and intractability: A guide to the theory of np-completeness*, San Francisco, CA: Freeman (1979).
- [13] D. Golovin and Q. Zhang, *Random hypervolume scalarizations for provable multi-objective black box optimization*, ICML2021 (2021).
- [14] A.A. Hagberg, D.A. Schult, and P.J. Swart, *Exploring network structure, dynamics, and function using networkx*, Proceedings of the 7th Python in Science Conference, Pasadena, CA USA, 2008, pp. 11 – 15.
- [15] Y. Haimes, L. Lasdon, and D. Wismer, *On a bicriterion formulation of the problems of integrated system identification and system optimization*, IEEE Trans. Syst., Man, Cybernetics **1** (1971), 296–297.
- [16] P. Hansen, *Bicriterion path problems*, Fandel, G., Gal, T. (eds) Multiple Criteria Dec. Making Theory Application. Lecture Notes Econ. Math. Syst. **177** (1980), 109–127.
- [17] C.L. Hwang and A.S.M. Masud, *Multiple objective decision making, methods and applications: a state-of-the-art survey*, Springer-Verlag (1979).
- [18] J. Isbell and W. Marlow, *Attrition games*, Naval Res. Logist. Quart **3** (1956), 71–94.
- [19] F. Kelly, A. Maullo, and D. Tan, *Rate control for communication networks : shadow prices, proportional fairness and stability*, J. Oper. Res. Soc. **49** (November 1997).
- [20] J. Larusic and A.P. Punnen, *The balanced traveling salesmanproblem*, Comput. Oper. Res. **38** (2011), 868–875.
- [21] S. Martello, W. Pulleyblank, P. Toth, and D. de Werra, *Balanced optimization problems*, Oper. Res. Lett. **3** (1984), 275–278.
- [22] F. Mornati, *Pareto optimality in the work of pareto*, Eur. J. Social Sci. **51-2** (2013), 65–82.

- [23] M.H. Nguyen, M. Baiou, and V.H. Nguyen, *Nash balanced assignment problem*, Proc. Int. Symp. Combinatorial Optim. (ISCO 2022).
- [24] M.H. Nguyen, M. Baiou, V.H. Nguyen, and T. Vo, *Nash fairness solutions for balanced tsp*, Proc. Int. Network Optim. Conference (INOC 2022), 93–98.
- [25] W. Ogryczak, H. Luss, M. Pioro, D. Nace, and A. Tomaszewski, *Fair optimization and networks: A survey*, J. Appl. Math. (2014), 1–26.
- [26] R.D. Plante, *The nozzle guide vane problem*, Oper. research **36** (1988), 18–33.
- [27] A. Raith and M. Ehrgott, *A comparison of solution strategies for biobjective shortest path problems*, Comput. Oper. Res. **36** (2009), 1299–1331.
- [28] A. Raith, M. Schmidt, A. Schöbel, and L. Thom, *Extensions of labeling algorithms for multi-objective uncertain shortest path problems*, Networks **72** (2018), 84–127.
- [29] M.A. Ramos, B. Marianne, M. Ludovic, and D. Serge, *Multiobjective optimization using goal programming for industrial water network design*, Ind. Eng. Chem. Res **53** (2014), 17722–17735.
- [30] G. Reinelt, *Tsplib a traveling salesman problem library*, ORSA journal computing **3** (1991), 376–384.
- [31] C. Sen, *A new approach for multi-objective rural development planning*, Indian Econ. J. **30** (1983), 91–96.
- [32] A.J. Skriver and K.A. Andersen, *A label correcting approach for solving bicriterion shortest-path problems*, Comput. Oper. Res. **27** (2000), 507–524.
- [33] A.J.V. Skriver, *A classification of bicriterion shortest path (bsp) algorithms*, Asia-Pacific J. Oper. Res. **17** (2000), 199–212.
- [34] A.P. Wierzbicki, *A mathematical basis for satisficing decision making*, Math. Modelling **3-5** (1982), 391–405.
- [35] H. Yarchow, *Locally convex spaces*, Stuttgart: B.G. Teubner (1981), 47–55.
- [36] L. Zadeh, *Optimality and non-scalar-valued performance criteria*, IEEE Trans Automat Contr **AC-8** (1963), 59–60.
- [37] Z. Zeitlin, *Minimization of maximum absolute deviation in integers*, Discr. Appl. Math. **3** (1981), 203–220.

APPENDIX

| Appendix 1: Young's inequality for products

We restate the proof of Young's inequality for products as follows.

Theorem 15 Suppose $a, b > 0, \alpha, \beta \geq 0$ and $\alpha + \beta = 1$ then $\alpha a + \beta b \geq a^\alpha b^\beta$.

Proof Since the logarithm function is concave, using Jensen's inequality we have

$$\log(\alpha a + \beta b) \geq \alpha \log(a) + \beta \log(b) = \log(a^\alpha b^\beta),$$

Thus, $\alpha a + \beta b \geq a^\alpha b^\beta$. Furthermore, the equality holds if and only if $a = b$ or $(\alpha, \beta) \in \{(0, 1), (1, 0)\}$. □