Weighted aggregation of partial rankings using Ant Colony Optimization

Peer-reviewed author version

NAPOLES RUIZ, Gonzalo; Falcon, Rafael; DIKOPOULOU, Zoumpolia; PAPAGEORGIOU, Elpiniki; Bello, Rafael & VANHOOF, Koen (2017) Weighted aggregation of partial rankings using Ant Colony Optimization. In: NEUROCOMPUTING, 250, p. 109-120.

DOI: 10.1016/j.neucom.2016.07.073

Handle: http://hdl.handle.net/1942/23694

Weighted aggregation of partial rankings using Ant Colony Optimization

Gonzalo Nápoles[a,b], Rafael Falcon[c], Zoumpoulia Dikopoulou[a], Elpiniki Papageorgiou[d,a], Rafael Bello[b], Koen Vanhoof[a]

[a]*Faculty of Business Economics, Hasselt Universiteit, Belgium*
[b]*Department of Computer Science, Universidad Central de Las Villas, Cuba*
[c]*Electrical Engineering and Computer Science, University of Ottawa, Canada*
[d]*Department of Computer Engineering, Technological Education Institute of Central Greece, Greece*

Abstract

The aggregation of preferences (expressed in the form of rankings) from multiple experts is a well-studied topic in a number of fields. The Kemeny ranking problem aims at computing an aggregated ranking having minimal distance to the global consensus. However, it assumes that these rankings will be complete, i.e., all elements are explicitly ranked by the expert. This assumption may not simply hold when, for instance, an expert ranks only the top-K items of interest, thus creating a partial ranking. In this paper we formalize the *weighted Kemeny ranking problem for partial rankings*, an extension of the Kemeny ranking problem that is able to aggregate partial rankings from multiple experts when only a limited number of relevant elements are explicitly ranked (top-K), and this number may vary from one expert to another (top-K_i). Moreover, we introduce two strategies to quantify the weight of each partial ranking. We cast this problem within the realm of combinatorial optimization and lean on the successful Ant Colony Optimization (ACO) metaheuristic algorithm to arrive at high-quality solutions. The proposed approach is evaluated through a real-world scenario and 190 synthetic datasets from `www.PrefLib.org`. The experimental evidence indicates that the proposed ACO-based solution is capable of significantly

Email addresses: `gonzalo.napoles@uhasselt.be` (Gonzalo Nápoles), `rfalcon@uottawa.ca` (Rafael Falcon), `zoumpoulia.dikopoulou@uhasselt.be` (Zoumpoulia Dikopoulou), `epapageorgiou@mail.teiste.gr` (Elpiniki Papageorgiou), `rbellop@uclv.edu.cu` (Rafael Bello), `koen.vanhoof@uhasselt.be` (Koen Vanhoof)

outperforming several evolutionary approaches that proved to be very effective when dealing with the Kemeny ranking problem.

Key words: Kemeny ranking problem, partial rankings, weighted aggregation, swarm intelligence, ant colony optimization.

---

1. Introduction

The aggregation of preferences from multiple experts is a well-studied topic in a number of fields such as *economic theory* (properties of a social choice function under elevation of pairs) [1], *social choice theory* (preference aggregation from a small subset of critical nodes in social networks) [2], *multi-criteria decision making* (group decision making) [3], *machine learning* [4] (evolutionary voting in classifier ensembles), *multi-agent systems* (reaching consensus in high-dimensional linear systems) [5] or *computational biology* (consensus genetic mapping) [6].

When these preferences are elicited in the form of N *rankings* over M objects/items (where each ranking denotes the preference of a single expert), the goal is to build a *consensus* (aggregated) ranking that reflects the set of individual preferences as faithfully as possible. Several methods to aggregate the ranking preferences of multiple voters have been proposed in the literature [7] [8] [9] [10]. Arrow's axioms [11] state, however, that no aggregation method could simultaneously satisfy three fairness criteria: *non-dictatorship* (the voting results cannot simply mirror that of any single person's preferences without consideration of the other voters), *Pareto efficiency* (if every individual prefers a certain option to another, then so must the resulting societal preference order) and *independence of irrelevant alternatives* (changes in individuals' rankings of irrelevant alternatives –ones outside a certain subset– should have no impact on the societal ranking of the subset).

In spite of the above result, it is still possible to compute an *aggregated ranking* having minimal distance to the global consensus. This ranking is referred to as the *Kemeny ranking* [12] [13] and interpreted as a maximum likelihood estimator of the "correct" ranking. Unfortunately, the Kemeny ranking is NP-hard to calculate. The study in [14] thoroughly investigates different optimization methods (exact and approximate algorithms) for computing the Kemeny ranking. The authors concluded that heuristic approaches are recommended in contexts having weak or no consensus. More recently, Aledo et. al. [15] resorted to evolutionary algorithms to come to grips with this challenging problem. Their results outperformed the remaining tested algorithms. Nevertheless, the proposed model is thought of for *complete rankings* (i.e., those in which each element is explicitly ranked) and cannot be directly applied to the aggregation of incomplete (*partial*) preferences, i.e., where only a subset of the available items is explicitly ranked. In this paper, we investigated two types of partial rankings that could be described as follows:

1. **Top-K rankings**: All respondents exactly select K relevant factors, whereas the remaining factors are placed at the K+1 position. In this kind of partial ranking, ties into the top-K ranked positions are not allowed.
2. **Top-K_i rankings**: Each respondent R_i is free to select K_i relevant factors that may be partially or completely ordered, whereas the remaining factors are placed at the K_i+1 position. In this scenario, tied factors into the top-K_i ranked positions could be observed.

This paper brings forth the following contributions. (1) We address the weighted aggregation of the two previous types of partial rankings from multiple experts by formulating the *weighted Kemeny ranking problem for partial rankings*, an extension of the Kemeny ranking problem that is able to aggregate top-K and top-K_i partial rankings from multiple experts. (2) We cast this problem into the realm of combinatorial optimization and lean on Ant Colony Optimization (ACO) [16], one of the most popular Swarm Intelligence [17] schemes, as the underlying optimization engine. In this scheme, we proposed two improved rules to compute the heuristic information used by ants to select the next state. (3) We introduce two heuristic strategies to derive the weight of each partial ranking in presence of subjective expert information (i.e., a set of predefined categories) or in its absence. In the first strategy, the weight is calculated from the fuzzy membership grade of each partial ranking to a set of predefined categories. If these predefined categories are not available, then the weight is computed as the ratio of non-tied items included in the partial ranking. (4) We conduct an extensive empirical analysis by comparing our solution against 11 other methods (two simple greedy techniques and 9 evolutionary optimizers) using a real-world scenario wherein Belgian respondents rank different aspects of potential employers, and 190 synthetic datasets taken from `www.PrefLib.org`. The empirical evidence indicates that the ACO-based approach is capable of significantly outperforming the other models for datasets under consideration.

The rest of the article is organized as follows. Section 2 briefly examines the Kemeny ranking problem and discusses several methods for aggregating partial rankings. Section 3 elaborates on a weighted extension of the Kemeny rule for aggregating partial rankings while Section 4 goes over the ACO fundamentals and revisits the three most prevalent models. Section 5 is concerned with tailoring ACO to solve the weighted Kemeny ranking problem, including the learning of the heuristic information matrix from the available data. Two heuristic strategies to compute the weight of each partial ranking are described in Section 6. The empirical study carried out to validate the proposed approach is unveiled in Section 7. Conclusions and future work directions are outlined in Section 8.

2. Related work and remaining challenges

This Section briefly reviews relevant works related to voting rules, the Kemeny ranking problem for complete rankings as well as other approaches for the aggregation of partial rankings.

2.1. Voting rules and Kemeny ranking problem for complete rankings

A *voting rule*, a.k.a *rank aggregation rule*, takes as input multiple rankings over the same element set and produces as outcome either a single element (the winner) or a consensus ranking of these elements [13].

Among the many different voting rules proposed in the literature [18], the *plurality rule* is perhaps one of the best known and most often applied *scoring rules*. This rule ranks items by the frequency with which they are placed first in the rankings. One may notice that other important considerations present in each ranking are simply disregarded by this procedure.

The *Borda rule* is another scoring rule. Each candidate earns as many points as the number of candidates ranked lower than himself. The winner is the one with the most points.

The *single transferable vote rule* goes through a series of $M-1$ rounds, each one eliminating the element with the lowest plurality score from every ranking. The last remaining element is the winner.

The *Bucklin rule* computes a score for each element that is based on the number of voters that ranked it among the top-K candidates. An element "passes the post" if it is selected within the lowest K elements by at least half of the voters. Ties are broken by the number of votes by which the post is passed.

The *maximin rule* ranks elements after a score based on pairwise counts of the number of votes that placed that element higher than another element.

The *Copeland rule* also follows a score but this time an element earns/loses a point for every pairwise election it wins/loses.

The *ranked pairs rule* also returns a ranking based on an ordering of all element pairs (a,b) according to the number of voters that prefer a over b.

Another well-studied rule is the *Kemeny rule* [12] [15], which operates on complete rankings. This rule yields a ranking that maximizes the number of pairwise agreements among the individual rankings (votes), where a pairwise agreement is reached whenever the ranking agrees with one of the votes on which a pair of candidates is ranked higher [13]. More formally, given a set of N rankings $X = \{X_1, X_2, ..., X_N\}$ over M elements, the *Kemeny ranking problem* is concerned with finding the ranking X_* that satisfies Equation (1), where $\mathcal{P}$ stands for the set of all possible permutations over M elements (there are $M!$ possible permutations) and $\mathcal{K}(X_i, Y)$ denotes the Kendall-Tau distance between X_i and Y. The resultant ranking X_* is called the *Kemeny ranking* of the set and construed as the one minimizing the number of disagreements among all rankings in X [15].

$$X_* = \underset{Y \in \mathcal{P}}{\operatorname{argmin}} \frac{1}{N} \sum_{i=1}^{N} \mathcal{K}(X_i, Y) \tag{1}$$

2.2. Methods for aggregating partial rankings

González-Pachón and Romero [19] approach the aggregation of *quasi orders* (i.e., incomplete ordinal rankings) as a *consensus search* by using distance functions. Interval goal programming (IGP) is presented as their solver of choice that tries to establish a weak consensus over incomplete ordinal rankings.

Klementiev et. al. [20] proposed a rank aggregation method for both permutations and top-K lists where they account for the *type* of the elements being ranked, i.e., they could belong to different data domains, so as to include the notion of *domain expertise*. Given only a set of constituent rankings, they learn an aggregation function that attempts to recreate the true ranking without labeled (type) data. The method is based on a mixture of distance-based models and leans on the Expectation-Maximization (EM) algorithm to estimate its parameters. The new technique significantly and robustly outperformed their previous domain-agnostic model [21].

Ammar and Shah [22] consider partial data in the form of *first-order* or *comparison* marginals. They treat this information as partial samples from an unknown distribution over permutations and provide an efficient algorithm for finding an aggregate complete ranking directly from the data without first learning the underlying distribution; this is an appealing feature for designing large-scale ranking systems such as *recommendation systems*.

Neghaban et. al. [23] remarked that the approach in [22] requires information about comparisons between all element pairs, and for each pair it requires the exact pairwise comparison marginal w.r.t the underlying permutation distribution. This assumption is not always easy to meet since, in reality, all pairs of items are not usually compared. The authors then propose an algorithm that takes as input the noisy comparison marginals for a subset of all possible item pairs and spits out scores for each item. The noise in the underlying permutation distribution is modeled after the Multinomial Logit (MLN) method [24]. Their algorithm has a natural *random walk* interpretation over the graph of objects with edges present between two objects if they are compared; the scores turn out to be the stationary probability of this random walk. The empirical analysis indicates that the proposed scheme performs comparably to the maximum likelihood estimator of the MLN model and outperforms the technique in [22].

Brandenburg et. al. [25] studied the aggregation of partial rankings under the nearest neighbor (NN) and Hausdorff versions of the Kendall-Tau distance. They proved that this problem is **NP-complete** under the NN Kendall-Tau distance even for two voters and that, in contrast, it is **NP-hard** and **coNP-hard** under the Hausdorff Kendall-Tau distance for at least four voters.

2.3. Remaining challenges

In spite of the Arrow's impossibility theorem, researchers continue addressing the aggregation of several preferences by solving the Kemeny ranking problem. Young and Levenglick [26] show that the Kendall distance (and consequently its extensions) is the only distance function ensuring the permutation(s) minimizing the Kemeny ranking problem have three desirable properties of being *neutral*, *consistent* and *Condorcet*. The Condorcet property means that, if there exists a permutation such that the order of every pair of elements is the order preferred by the majority, then that permutation has minimum distance to the voters' permutations. Therefore, the main challenge towards this goal lies on the performance of the discrete optimizer used when solving the related combinatorial problem. On the other hand, there exist situations for which rankings are partial and therefore, the classical Kendall-Tau distance is no longer suitable.

The second challenge refers to the inclusion of the weighted approach when aggregating partial rankings and the automatic estimation of the membership degree of a ranking to the population. Recently Nápoles et al. [27] proposed a two-step methodology to build fuzzy prototypes from a population of partial rankings. Being more explicit, in the first step the authors put forth a fuzzy clustering algorithm for partial rankings called fuzzy c-aggregation, while the second step is focused on solving the extended Kemeny ranking problem for each discovered cluster taking into account the estimated partition matrix. Despite the novelty of this approach, the reader may notice that this algorithm will produce c different aggregations, with c being the number of clusters detected by the clustering algorithm. However, the clustering approach may not be adequate for some scenarios where a single aggregated solution is expected. This implies that other approaches to compute the membership degrees of partial rankings are required.

3. Weighted aggregation of partial rankings

In this section we extend the well-known Kemeny ranking problem [12] by considering that orderings to be clustered may be incomplete or partial (i.e., tied elements are allowed). Besides, we assume that each partial ordering X_i has an associated weight $\omega_i \in [0, 1]$ representing the extent to which the ranking belongs to the population. Formally, *the weighted Kemeny ranking problem for partial rankings* could be summarized as follows. Let $X = \{X_1, ..., X_i, ..., X_N\}$ be a set of N partial rankings over M items $F = \{F_1, ..., F_l, ..., F_M\}$ where the ith ranking comprises the vote of a single respondent with weight $\omega_i \in [0, 1]$. More explicitly, we can describe a partial ranking X_i as a vector $\{X_i^1, ..., X_i^k, ..., X_i^M\}$ where $X_i^k \preceq X_i^{k+1}$ denotes that X_i^k precedes X_i^{k+1}. The theoretical challenge is to construct a fair enough ranking Y taking into account all input (potentially partial) rankings and their weights. It should be mentioned that the solution for the weighted Kemeny ranking problem is a complete ranking, and therefore ties are not allowed.

Being more explicit, the solution for the weighted Kemeny problem is equivalent to computing a complete ranking with minimal distance to the global consensus. Equation (2) formalizes the objective function to be minimized, where $\mathcal{H}(X_i, Y)$ represents a distance function quantifying the dissimilarity between the ith partial ranking and the candidate solution Y to be evaluated.

$$\min \to \mathcal{F}(Y) = \sum_{X_i \in X} \omega_i \mathcal{H}(X_i, Y) \Big/ \sum_i \omega_i \tag{2}$$

The reader may notice that the first modification to the standard Kemeny ranking problem lies on the inclusion of the weight quantifying the extent to which the ith ranking belongs to the population. The second modification is related to the normalized distance function $\mathcal{H}(.,.)$ to compute the dissimilarity degree between two rankings with tied elements. Notice that the standard Kemeny ranking problem uses the Kendall-Tau distance [28], which measures the dissimilarity as the number of item pairs over which the two rankings disagree. However, the original Kendall-Tau distance is no longer adequate when comparing rankings having tied items since this distance assumes that items are all ordered. Instead, we could adopt other versions of the Kendall-Tau distance or other extended dissimilarity measures such as the Hausdorff distance [29], the Spearman's footrule distance [30] or the Goodman-Kruskal's one [31]. Having several metrics for partial rankings is obviously convenient, but it poses the question of which one would be better suited when comparing partial rankings when solving the Kemeny ranking problem.

The Goodman-Kruskal's approach is not always defined and thus there could be scenarios where this procedure fails. Moreover, Fagin et al. [32] mathematically proved that the Hausdorff variants of the Kendall-Tau distance and the Spearman's footrule distance are actually equivalent. This outcome was based on the Diaconis-Graham inequality [33], which asserts that the Kendall-Tau distance and the Spearman's footrule distance are within a factor of two from each other. It implies that selecting a distance function does not matter so much when solving the weighted Kemeny ranking problem, as long the distance function is capable to deal with partial rankings.

In this paper we use the Hausdorff version of the Kendall-Tau distance as the dissimilarity functional when aggregating partial rankings. The Hausdorff distance has been extensively studied and shown to have particularly flexible mathematical and algorithmic properties [32]. Equation (3) formalizes this distance, where X_i is the ith partial ranking, Y denotes the Kemeny ranking to be evaluated, $\mathcal{K}(X_i, Y)$ is the set of all item pairs that appear in different order, $\mathcal{R}_1(X_i, Y)$ is the set of all item pairs which are tied in X_i but not tied in Y, while $\mathcal{R}_2(X_i, Y)$ is the set of all item pairs which are tied in the ranking Y but not tied in the ith partial ranking. This function can also be adopted for comparing full rankings, thus leading to the Kendall-Tau distance (i.e., $\mathcal{R}_1(X_i, Y) = \mathcal{R}_2(X_i, Y) = 0$).

$$\mathcal{H}(X_i, Y) = |\mathcal{K}(X_i, Y)| + \max\{|\mathcal{R}_1(X_i, Y)|, |\mathcal{R}_2(X_i, Y)|\} \tag{3}$$

The inclusion of the Hausdorff distance $\mathcal{H}(X_i, Y)$ in the objective function (2) allows computing the dissimilarity between each input ranking and the candidate (complete) aggregated ranking. Due to the fact that Y is a full ranking, we could compute $\mathcal{H}(X_i, Y) = |\mathcal{K}(X_i, Y)| + |\mathcal{R}_1(X_i, Y)|$. This is possible because there are no tied items in a full ranking (i.e., $|\mathcal{R}_2(X_i, Y)| = 0$). On the other hand, the reader may verify that $|\mathcal{R}_1(X_i, Y)| = \binom{M-K_i}{2}$ where K_i is the number of relevant items selected by the ith respondent. Notice that we assume partial rankings with non-homogeneous tied factors, since there are scenarios where each respondent may select a different number of relevant items. Equation (4) shows the normalized objective function to be optimized.

$$\min \rightarrow \mathcal{F}(Y) = \left(\sum_{X_i \in X} \frac{2\omega_i \left[|\mathcal{K}(X_i, Y)| + \binom{M-K_i}{2} \right]}{M(M-1)} \right) \Big/ \sum_i \omega_i \tag{4}$$

Equation (4) involves a **NP-hard** problem with a search space comprised of $M!$ possible states (i.e., the set of all permutations over M items). In order to deal with the computational intractability of this weighted aggregation problem, Nápoles et al. [34] proposed a novel approach based on Swarm Intelligence that exploits a colony of artificial ants. However, this approach does not take into account the weight of partial rankings. Recently, Nápoles et al. [27] extended the crisp method in order to construct prototypes from fuzzy information granules discovered by a clustering algorithm. Before describing the details of this procedure, next we provide a basic background about Ant Colony Optimization that will be used to solve the weighted Kemeny ranking problem formulated before.

4. Ant Colony Optimization

The generation of feasible permutations representing complete rankings is entrusted in this study to the ACO methods. The objective function in Equation (4) evaluates the quality of each candidate solution (ant tour) during the search process.

The ACO metaheuristic is a biologically-inspired search technique that was originally devised to solve combinatorial optimization problems [16]. Its creator, Marco Dorigo, drew inspiration from the manner in which ants corporately forage. They depart from the nest and once a source of food is identified, they deposit a chemical substance on the ground named *pheromone* on their way back to the nest; these pheromone trails serve to guide the rest of the colony towards the food source [35]. ACO is one of the hallmark swarm intelligence algorithms and bears a plethora of successful applications to real-world problems [36] [37] [38].

ACO is a fully constructive model where each ant builds a candidate solution to the problem by incrementally exploring the nodes (or edges) of a search graph. Each artificial ant moves from one state to another during the search process (here states are components of the solution). As depicted in Equation (5), the likelihood of moving from one node to another (ACO transition rule) at the next discrete time step $t+1$ mainly rests on two parameters: (1) the *collective information* $\tau_{kl}(t)$ derived from the pheromone trails and iteratively updated by ants during the navigation of the search graph and (2) the *heuristic information* η_{kl} denoting the invariant, problem-specific preference of moving from one state to another. The heuristic component must be carefully provided/estimated as it is treated as an invariant, i.e., it is not modified throughout the algorithm's execution.

$$P_{kl}^{v}(t+1) = \frac{[\tau_{kl}(t)]^{\alpha}[\eta_{kl}]^{\beta}}{\sum\limits_{r \in \mathcal{N}_k^v} [\tau_{kr}(t)]^{\alpha}[\eta_{kr}]^{\beta}},\ l \in \mathcal{N}_k^v \tag{5}$$

In light of the Kemeny ranking problem, Equation (5) denotes the probability of accepting the lth state (i.e., next factor to be ranked) at the kth position of the candidate ranking, $\mathcal{N}_k^v$ is the set of unvisited states (factors) at the kth position for the vth ant while α and β govern the strength of the collective and heuristic information, respectively.

Once the individual ant tours are completed, the pheromone levels on all trails using the solutions found by the agents will be updated. First, pheromone evaporation takes place uniformly thus reducing the amount of pheromone on all trails. Subsequently, certain pheromone amount will be added to the nodes/edges of the more promising solution(s). This is a very important step in any ACO-based implementation; most of ACO variants differ mainly in the strategy used for updating the collective information (pheromone trails) at each iteration. In the sequel we will discuss the three most popular ACO algorithms.

4.1. Ant System

Ant System (AS) is credited with being the first ACO algorithm [39]. The pheromone trails are updated once all ants have completed their tours. A certain portion of the pheromone in each trail is evaporated according to a factor $0 < \rho < 1$. Afterwards, each ant v deposits a pheromone amount $\Delta\tau_{kl}^{v}$ proportional to the quality of its solution along the edges belonging to it. This pheromone update rule is reflected in Equation (6), with S being the number of ants in the colony.

$$\tau_{kl}(t+1) = (1-\rho)\tau_{kl}(t) + \sum_{v=1}^{S} \Delta\tau_{kl}^{v} \tag{6}$$

The long-term effect of the above rule is that edges not frequently chosen by the ants will see their pheromone concentration gradually vanish whereas those edges selected by the ants will receive a boost in their pheromone amount, thus becoming more probable candidates for selection in future iterations. A more thorough study [39] revealed that better results could be attained if the pheromone increase is only applied by the global best solution rather than having all colony members do so. In spite of that, AS suffers from stagnation (convergence to local optima) due to the unbounded accumulation of pheromone over the best found edges.

4.2. Ant Colony System

Ant Colony System (ACS) improves the AS scheme by exploiting the global best solutions found during the search stage [40]. As a result, the algorithm exhibits superior exploitation features as ants build their solutions, instead of exploring new areas of the solution space. This goal is achieved via a three-fold mechanism: (1) a strong *elitist* strategy for updating pheromone trails, (2) a modification to the pheromone update rule and (3) a pseudo-random rule for selecting new states.

ACS' pheromone update rule is reported in Equation (7), with $\tau^*_{kl}(t)$ denoting the pheromone amount associated with the ant featuring the best heuristic value at time step t. Like in AS, pheromone evaporation affects all edges yet the boost is only reserved for those edges belonging to the best solution.

$$\tau_{kl}(t+1) = (1-\rho)\tau_{kl}(t) + \rho\tau^*_{kl}(t) \tag{7}$$

ACS' pseudo-random proportional rule in Equation (8) aims at fostering exploitation of the knowledge attained by the colony. In a nutshell, if a random number $q \sim U(0,1)$ falls below q_0 then the ant will move to the state maximizing the product between collective and heuristic information, otherwise ACS will adopt the standard decision rule in Equation (5). Notice that q_0 is a user-defined parameter that favors exploitation over exploration as it approaches 1.

$$l = \underset{r \in \mathcal{N}_k^v}{\operatorname{argmax}} \left\{ [\tau_{kl}(t)]^\alpha [\eta_{kl}]^\beta \right\} \text{ if } q \leq q_0 \tag{8}$$

The third distinctive element in the ACS model is the iterative pheromone update rule ants employ as they build their solution, as shown in Equation (9). This approach has the same effect as decreasing the probability of selecting the same path for all ants, thus introducing a balance between exploitation and exploration.

$$\tau_{kl}(t+1) = (1-\rho)\tau_{kl}(t) + \rho\tau^*_{kl}(0) \tag{9}$$

The ACS algorithm frequently reports better performance than AS owing to its emphasis on the exploitation of the most promising solutions discovered by the colony.

4.3. MAX-MIN Ant System

Like ACS, the *MAX-MIN Ant System* (MMAS) [41] was specifically engineered to pursue a stronger exploitation of solutions, thus avoiding the stagnation problems encountered by AS. This model has the following features: (1) similar to ACS, a strong elitist strategy regulates the ant allowed to update the pheromone trails (either the best-so-far ant or the one with the best solution in the current iteration); (2) all pheromone trails are bound to the range $[\tau_{MIN}, \tau_{MAX}]$. If $\tau_{MIN} > 0$ for all solution components, then the probability of choosing a specific state will never be zero, which avoids stagnation configurations. Finally, pheromone trails are initialized with τ_{MAX} to ensure further exploration of the search space at the beginning of the optimization phase.

MMAS has also reported very encouraging results in the literature, even outperforming ACS [42] [36].

5. Solving the weighted Kemeny ranking problem

In this section we explain how to optimize the objective function defined in Section 3 by exploiting a colony of artificial ants. With this goal in mind, we defined four central components:

- The structure of the pheromone graph used by ants to construct the solutions.
- The interpretation of the probabilistic rule to select the next state.
- The formal definition of the set of feasible states at each step.
- The estimation of the heuristic information.

As mentioned, the goal of the search method is to produce a complete ranking (i.e., a permutation over M different factors) minimizing the objective function (4). This problem is similar to the well-known Traveling Salesman Problem [43] where artificial agents construct the candidate solution by traveling along a fully connected graph. The graph nodes correspond to the M elements $F = \{F_1, ..., F_l, ..., F_M\}$ to be ordered. Due to the fact that a solution to the weighted Kemeny ranking problem is a permutation of such M items, each item F_l will appear exactly once. This suggests that self-connected graph nodes are not allowed, otherwise the Kemeny ranking may involve explicit tied items. However, a Kemeny ranking might comprise implicit tied items (i.e., items that may be freely exchanged without altering the heuristic value) which is a result of frequent ties over the same two items.

In the proposed scheme, the transition value P_{kl} is the probability of accepting the lth item at the kth ranking position. This approach is slightly different from other scenarios where the transition value P_{kl} often denotes the probability of moving to the lth graph node from the kth node. In practice, both approaches are equivalent because the probability of accepting the lth ranking item at the kth position will eventually be influenced by those ranking items situated at the previous $(k-1)$ positions. From this remark we can formally define the set of feasible states for the vth ant at each step k. The domain set $\mathcal{N}_k^v \subseteq F$ is given by $\mathcal{N}_k^v = \{F_1, ..., F_l, ..., F_M\} - \{Y_v^1, Y_v^2, ..., Y_v^{k-1}\}$ where Y_v^{k-1} represents the item located at the $(k-1)$ ranking position, according to the vth agent. Being more explicit, all previously ranked items are no longer part of the neighborhood of the ant at the kth step. This ensures the unicity of ranking items in the solution for the Kemeny ranking problem.

Another important aspect when solving combinatorial problems using ACO-based algorithms is the estimation of the heuristic matrix. The accurate estimation of the heuristic component often leads to high-quality solutions, otherwise the solutions to the weighted Kemeny ranking problem will probably be sub-optimal. In the following sections we propose two strategies to estimate the heuristic matrix from input data, assuming two partial ranking aggregation scenarios.

5.1. Weighted aggregation of multiple top-K rankings

The first scenario takes place when each respondent selects the top-K items. It implies that each input ranking will be partial in the sense that only the top-K items are explicitly ranked, whereas the other $M-K$ items are tied at the $K+1$ position. It can be noticed that estimating the heuristic values for the M items across the first K positions is equivalent to computing the number of observations on which the lth element *was observed* at the kth position $(k = 1, 2, ..., K)$. For the $M-K$ last positions, this heuristic cannot be directly used since such items are tied. However, we may compute the number of observations on which the item *was not included* into the top-K.

Equation (10) formalizes the above reasoning, where $\vartheta_k(F_l)$ denotes the sum of the weights of those rankings on which item F_l was ranked at the kth position $(1 \leq k \leq K)$, while $\sim \vartheta_k(F_l)$ is the sum of the weights of those rankings on which the lth item was not included into the top-K. For the latter case divide the expression by $M-K$ since these items have probability $(M-K)/M$ to be placed at the last positions, assuming a uniform probability distribution.

$$\eta_{kl} = \begin{cases} \vartheta_k(F_l)\left(\sum_i \omega_i\right)^{-1} & , k \le K \\ \dfrac{\sim \vartheta_k(F_l)}{M-K}\left(\sum_i \omega_i\right)^{-1} & , k > K \end{cases} \tag{10}$$

It should be specified that the functions $\vartheta_k(F_l)$ and $\sim \vartheta_k(F_l)$ must consider the fact that X_i belongs to the ranking population with weight ω_i. Equation (11) show how to compute the function $\vartheta_k(F_l)$, but it may be easily extended for $\sim \vartheta_k(F_l)$. On the other hand, in Equation (10) the sum of all weights $\sum_i \omega_i \le N$ is used to normalize the heuristic values.

$$\vartheta_k(F_l) = \sum_{X_i \in X} \begin{cases} \omega_i & , X_i^k = F_l \\ 0 & , X_i^k \ne F_l \end{cases} \tag{11}$$

Example 1. Let us consider a weighted aggregation scenario of $N = 5$ partial rankings over $M = 5$ items where each expert selected the $K = 3$ most relevant factors. Table 1 summarizes this scenario, where each row involves a partial ranking. According to Equation (10), the heuristic preference of accepting the item F_2 at the second position is given by $\eta_{22} = \vartheta_2(F_2)/2.2 = 0.8/2.2 \approx 0.36$. Similarly we can compute the remaining components of the heuristic matrix.

Table 1: Example of a weighted aggregation of multiple top-K rankings.

	ω_i	F_1	F_2	F_3	F_4	F_5
X_1	0.8	1	2	$K+1$	3	$K+1$
X_2	0.2	2	1	$K+1$	$K+1$	3
X_3	0.5	1	3	$K+1$	$K+1$	2
X_4	0.1	1	3	$K+1$	2	$K+1$
X_5	0.6	2	1	3	$K+1$	$K+1$

Observe that according to Equation (10) some heuristic values could be zero (e.g. $\eta_{25} = 0$ since F_2 was always included into the top-3) and therefore the probability P_{kl}^v of selecting these states will be zero. However, normally the probability value P_{kl}^v should not be zero since it is possible to estimate a good solution having F_2 at the last position. In order to overcome this issue we replace all zero-values by $\eta_{MIN} = \min\{\eta_{kl}\}$ such that $\eta_{kl} \ne 0$, thus we guarantee that all states have a probability to be visited by artificial ants, although they have less chance to be selected.

5.2. Weighted aggregation of multiple top-K_i rankings

This scenario is more complex (but also more informative) because each respondent is free to select K_i items such that $2 \leq K_i \leq M$. Since the number of relevant items could change from a respondent to another, we cannot simply count the number of observations of each item. In order to compute a more realistic heuristic matrix we may compute the relative frequency on which an item could be observed at each ranking position. Notice that this assumption attempts to hypothetically break the ties in order to transform partial rankings into complete ones. Equation (12) enunciates this method, where $Q_{K_i}(F_l)$ is the set of all rankings where item F_l was excluded from the top-K_i, $\psi_{X_i}(F_l)$ is the set of all feasible positions for the lth item, while $\vartheta_k(F_l)$ denotes the sum of the weights of those rankings on which item F_l was placed at the kth ranking position.

$$\eta_{kl} = \left(\vartheta_k(F_l) + \sum_{X_i \in Q_{K_i}(F_l)} \begin{cases} \omega_i & , k \in \psi_{X_i}(F_l) \\ 0 & , k \notin \psi_{X_i}(F_l) \end{cases} \right) \left(\sum_i \omega_i \right)^{-1} \tag{12}$$

Example 2. Let us consider the weighted aggregation scenario summarized in Table 2, with $N = 5$ partial rankings over $M = 5$ items where the ith respondent selected the most relevant K_i items. According to Equation (12), the heuristic value of accepting F_2 at the first position η_{12} is $1/2.2(0.7) \approx 0.318$ since $\vartheta_1(F_2) = 0.7, Q_{K_i}(F_2) = \{X_1, X_4\}, \psi_{X_1}(F_2) = \{4, 5\}, \psi_{X_4}(F_2) = 3, 4, 5$. Observe that item F_2 could not be hypothetically located at the first ranking position without introducing new tied pairs of items because $k \notin \psi_{X_1}(F_2) \cup \psi_{X_4}(F_2)$. This suggests that the heuristic value η_{12} is computed from the number of times F_2 was observed at the first position.

Similarly to the first scenario (i.e., respondents select the most relevant K items), we must avoid zero-values in the heuristic matrix, although this situation is possible (i.e., the item was never observed in a position and there is no chance to be observed without inducing new ties). However, it is still possible to build a candidate solution with this feature having minimal distance to the consensus, and therefore it must be considered as well. In these scenarios the probability should not be zero but rather small, e.g., $\eta_{MIN} = \min\{\eta_{kl}\}$

Table 2: Example of a weighted aggregation of multiple top-K_i rankings.

	ω_i	F_1	F_2	F_3	F_4	F_5
X_1	0.8	1	K_1+1	2	K_1+1	3
X_2	0.2	3	1	K_2+1	2	K_2+1
X_3	0.5	2	1	4	5	3
X_4	0.1	2	K_4+1	K_4+1	K_4+1	1
X_5	0.6	5	3	1	2	4

where $\eta_{kl} \neq 0$. This approach is similar to the thresholding strategy used in MAX-MIN Ant Systems [41] which proved to be quite effective in promoting the exploration of alternative regions of the search space. Next we propose two strategies to compute the weight of each partial ranking.

6. Heuristics to determine the weight of each partial ranking

A pivotal issue when solving the weighted aggregation problem is related to the estimation of weights. Strategies for weighting partial rankings could vary from expert-based estimations to more automated measures. In this paper we addressed this issue by considering two scenarios. In the first one, the weight is calculated from the fuzzy membership grade of each partial ranking to a set of predefined categories. If these predefined categories are not available (second scenario), then the weight is computed as the ratio of non-tied items included in the partial ranking. This latter heuristic is based on the fact that partial rankings having a fewer number of tied factors are more informative when solving the weighted aggregation problem, as they better express the user preferences. Note however that we are assuming that partial ranking instances have the same confidence level (i.e., all experts responses are equally reliable).

Next we describe an algorithm to compute the membership degree of each partial top-K or top-K_i ranking across a set of predefined categories. In this paper, we assume that a category is a disjoint set of items that comprises an information granule regularly defined by domain experts. This *fuzzy allocation problem* could be formalized as follows. Let us suppose a set of N partial rankings $X = \{X_1, ..., X_i, ..., X_N\}$ over M items $F = \{F_1, ..., F_l, ..., F_M\}$ and a set of categories $C = \{C_1, ..., C_j, ..., C_P\}$ resulting from a partition of the item set. The fuzzy allocation problem is equivalent to compute the degree on which partial rankings belong to each predetermined category. This allocation problem is fuzzy in nature because a partial ranking may be associated with several categories at the same time but with different degrees.

Essentially, the proposed method computes the correspondence degree between items in the partial ranking and those items belonging to each category. Observe that we cannot compute this correspondence degree using a distance function (e.g., the Hausdorff distance) since categories are unordered sets and therefore there is no ordinal relation among category items. The method comprises four well-defined steps which are described below.

Step 1. Compute the intersection set $\Phi_{ij} = X_i \cap C_j$ between the partial ranking X_i and the category C_j. This step allows determining, for each partial ranking, the set of ranked items that additionally belongs to the jth category. Notice that this step does not consider the existence of an ordinal relation between pairs of items included into the top-K (or top-K_i).

Step 2. Compute the relative relevance $Z(F_l)$ of each factor $F_l \in \Phi_{ij}$ using its position $1 \leq R_{(F_l)} \leq K$ into the top-K_i (or top-K) items associated with the ith partial ranking. The relevance $Z_i(F_l) = [(K_i + 1) - R_{(F_l)}]/K_i$ provides a local measure to determine the degree of membership to each category. In the case of top-K partial rankings, $K_1 = \ldots = K_i = \ldots = K_N$ since all respondents have to select exactly K relevant items.

Step 3. Compute the weight $\tilde{\omega}_i^{(j)}$ of the ith partial ranking to the jth category. To accomplish that, we adopt Equation (13) for both top-K and top-K_i scenario, assuming that ψ is the normalization factor.

$$\tilde{\omega}_i^{(j)} = \sum_{F_l \in \Phi_{ij}} \frac{Z(F_l)}{\psi} \tag{13}$$

It should be remarked that, for top-K partial rankings, the number of relevant items K may be different from the cardinality of the category C_j. If $|C_j| < K$ then the degree to which the ith partial ranking belongs to the jth category will never be maximal because some ranking positions cannot be covered. On the other hand, if $|C_j| > K$ the degree to which the ith partial ranking belongs to the jth category will never be maximal either because some ranking items cannot be selected by respondents. Both scenarios are considered when normalizing the sum of all relevance degrees, which allows computing fair membership degrees.

Therefore, for top-K partial rankings, $\psi = \frac{-\min\{K,|C_j|\}[-2K+\min\{K,|C_j|\}-1]}{2K}$. This normalization factor represents the sum of the first $\min\{K, |C_j|\}$ relevance degrees. Being more explicit, this sum of relevance degrees is equivalent to computing the sum of the first K numbers $i/K, i = \{1, .., K\}$ minus the sum of the $K - \min\{K, |C_j|\}$ numbers $i/K, i = \{1, .., K\}$. It implies that the maximal value for the sum of the relevance degrees $Z(F_l), \forall F_l \in \Phi_{ij}$ is reached when $C_j \subseteq X_i$ and items contained in C_j are placed at the first K ranking positions. The normalization factor allows estimating realistic values and may be inferred from the following expression:

$$
\begin{aligned}
&\left(\sum_{i=1}^{K} i/K\right) - \left(\sum_{i=1}^{K-\min\{K,|C_j|\}} i/K\right) \\
&= \frac{K+1}{2} - \frac{(K - \min\{K, |C_j|\})(K - \min\{K, |C_j|\} + 1)}{2K} \\
&= \frac{-\min\{K, |C_j|\}[-2K + \min\{K, |C_j|\} - 1]}{2K}
\end{aligned}
$$

In the case of top-K_i partial rankings, the number of relevant items will likely differ from one partial ranking to another. More importantly, some of these items may be placed at the same ranking position (i.e., tied items into the top-K_i are allowed). This feature increases the uncertainty in the decision-making process and may lead to quite similar membership degrees. If the relevant items are uniformly selected from homogenous categories, then all membership degrees will probably tend to the membership value $1/|C|$. This configuration cannot be observed in the previous scenario since we assumed that the top-K items are rigorously ordered.

Another issue that arises here is that there is no restriction on the number of relevant items to be selected by the respondent when constructing the top-K_i partial ranking. This implies that a specific category can be entirely included into the top-K_i ranking. It could be possible to allocate all selected items at the same ranking level (e.g., selected items belong to the same category and they are equally relevant to characterize the concept under evaluation). In this case, the normalization factor $\psi = |C_j|$.

Step 4. After computing the above equation for each category, the final membership values are calculated as $\omega_i^{(j)} = \tilde{\omega}_i^{(j)} / \sum_j \tilde{\omega}_i^{(j)}$. This ensures that the sum of all membership values will be exactly one, which is an important property to be preserved in fuzzy environments.

7. Numerical simulations

In this section, we evaluate the proposed weighted aggregation approach across several evolutionary approaches that proved to be adequate solvers of the Kemeny ranking problem. With this goal in mind, we used both real-world and synthetic datasets having different features.

7.1. Benchmarking algorithms and parametric settings

In this section, we describe the algorithms selected for benchmarking purposes. Recently Aledo et al. [15] proposed a solution scheme based on Evolutionary Computation for the Kemeny ranking problem. Results have shown that evolutionary algorithms clearly outperformed other tested algorithms (i.e., Borda counting index, variants of the Branch and Bound algorithm, among others). The central feature of the Genetic Algorithms (GA) used to solve the Kemeny ranking problem relies on the search space characteristics. Instead of dealing with the standard binary representation, they adopted a permutation-based solution encoding. During the search progress, the chromosome population evolves according to three genetic operators: selection, crossover and mutation. The selection operator promotes high-quality individuals and does not depend on the solution representation, but on the fitness value.

Nevertheless, in permutation-based search spaces crossover and mutation operators must be carefully defined; otherwise non-feasible solutions may be produced. In this study, we used the operators discussed in [44] to solve the *Traveling Salesman Problem*. Once promising individuals have been selected, they are (randomly) organized in pairs. Over each pair we apply the crossover operation by using one of the following three operators:

- **POS.** Position-based crossover operator [45]. It starts by selecting a set of random positions such that the values for these positions are kept in both parents. The remaining positions are completed by using the relative order in the other parent.

- **OX1.** Order crossover operator [46]. It selects two cut points $1 \leq c_1 < c_2 \leq M$ and then, for every parent, the genetic segment between the cut points c_1 and c_2 is directly copied into the corresponding child. Then, starting from c_2, the remaining items are copied into that child following the relative order in which they appear in the other parent, onwards and moving to the first position once the end of the individual is reached. The same procedure is repeated in the other child, by exchanging the role between the parent and the child.

- **OX2.** Order-based crossover operator [45]. It randomly selects several positions for each parent. Items in non-selected positions are maintained in the child, while the selected ones are set according to the positions taken by these items in the other parent.

Once offspring are generated by crossover, a mutation operator is applied over each offspring with a given mutation probability. In [13] the authors adopted the following evolutionary operators:

- **ISM.** Insertion mutation operator [47]. It randomly chooses an element in the permutation, which is removed and reinserted in a different (randomly selected) position.

- **DM.** Displacement mutation operator [48]. It randomly selects a segment of items in the permutation, which are removed and reinserted in a randomly selected position.

- **IVM.** Inversion mutation operator [49]. The semantic of this genetic operator is quite similar to DM but the removed items are reinserted as a single block in reverse order.

The combinations of the above permutation-based crossover and mutation operators lead to nine GA-based optimizers. For such evolutionary approaches, we used a population of 200 individuals. The mutation probability is set to 0.1 whereas the crossover probability was fixed to 0.9. Observe that the crossover probability is notably higher than the mutation probability as suggested in [15]. The search process stops after 50 generations, leading to 10,000 evaluations of the objective function. Normally, the number of generations used in population-based algorithms is higher than the number of individuals. However, after a number of preliminary simulations we observed that, for the same number of generations, the GA-based algorithms reported better results by using a larger population and fewer generations.

In the case of ACO-based algorithms, we adopted the following parameters: the number of ants was taken as the number of items to be ranked multiplied by $S = 3$; $\alpha = 2$ and $\beta = 3$, the pheromone matrix was initialized to $\tau_{kl}(0) = 0.5$ and the evaporation factor was set to $\rho = 0.8$. For the ACS algorithm, the parameter q_0 was initialized to 0.6 whereas for MMAS the pheromone thresholds τ_{MIN} and τ_{MAX} are computed as suggested in [41]. Finally, the search stops once the algorithm reaches 9,000 objective function evaluations. Notice that we reduced the number of evaluations allowed for ACO-based methods, since estimating the heuristic information requires further calculations.

In the above parameter configuration, $\beta > \alpha$ since the heuristic matrix provides a suitable information source when aggregating partial rankings, which is based on the principle of greedy aggregation methods. On the other hand, the homogeneous initialization of the heuristic matrix allows guiding the search mainly based on the heuristic information at the first iterations. However, solely promoting the heuristic information may lead to stagnation or premature-convergence configurations. Aiming at preventing these undesirable states, we selected a large evaporation factor.

Moreover, we include two simpler algorithms as baselines. The first one is the *weighted Borda counting* method[13], which computes a score for each item based on its position across all observations. Next, items are arranged according to their scores. The second baseline method, baptized as the *Greedy Ant Model* (GAM), relies on a single ant in ACS that systematically only exploits the heuristic knowledge to select the next feasible state. In GAM, $\alpha = 0$, $\beta = 1$ and $q_0 = 1$. Overall, we compare our approach against two greedy methods as baseline techniques and nine evolutionary algorithms.

7.2. Numerical simulations for a real-world dataset

In this section, we evaluate the proposed methodology by using a real study case concerning to the attractiveness of companies in Belgium [50][34][27]. During the data acquisition phase, 14585 Belgian respondents (aged between 18 and 65 years old) were consulted regarding two different ranking scenarios. Both surveys were conducted by a panel of marketing experts from Randstad[1] and are summarized as follows:

- **Scenario 1.** Each respondent ranked the top-5 factors out of $M = 17$ possible factors. From this survey we obtained 14,585 partial rankings where only the top-K factors are ordered, whereas the other $M - K$ factors are tied at ranking position $K + 1$.

[1] Randstad (`http://www.randstad.com`) is the second largest Human Resources (HR) provider in the world. It expanded its operations to 39 countries, representing more than 90 percent of the global HR services market.

- **Scenario 2.** In the second study each expert is free to select the most relevant K_i factors, such that $2 \leq K_i \leq M$, therefore the number of ranked elements is not necessarily homogeneous in all cases. More explicitly, in the ith ranking the top-K_i factors are ordered, whereas the remaining $M - K_i$ factors are ranked at the position $K_i + 1$. Moreover, in this kind of partial rankings, tied factors into the top-K_i ranked positions could be observed.

Solving these ranking aggregation problem allows characterizing the attractiveness of companies in Belgium, that is, their ability to attract highly-competent and productive employees. If workers prefer some factors (e.g. comfort, salary) when they are looking for an employer, and the evaluated company does not offer such features, then it is expected that more competent employees end up not working with that company. With this knowledge at hand, the company board may improve its branding and enhance its visibility which frequently results in better incomes. Table 3 displays the global factors (ranking items) evaluated by respondents that came up after a panel discussion of marketing experts.

Table 3: Global factors evaluated by each respondent during the online survey.

F_1	Financially sound
F_2	Offers quality training
F_3	Offers long-term job security
F_4	Offers international / global career
F_5	Future prospects / career opportunities
F_6	Strong management
F_7	Offers interesting jobs (job description)
F_8	Pleasant working environment
F_9	Competitive salary package
F_{10}	Good balance between life and work
F_{11}	Conveniently located
F_{12}	Strong image / pursues strong values
F_{13}	Quality products / services offered
F_{14}	Deliberately handles the environment and society
F_{15}	Uses the latest technologies / innovative
F_{16}	Provides flexible working conditions
F_{17}	Encourages diversity (age, gender, ethnicity)

Table 4 displays the list of expert-defined categories, which allows computing the weight of each partial ranking as explained in Section 6. Particularly, we adopted the heuristic strategy for predefined categories where factors are gathered according to their semantics by marketing experts.

Table 4: Categories determined by marketing experts.

	Name	Factors in the category
C_1	Salary	F_9
C_2	Stability	F_1, F_3
C_3	Future	F_2, F_5, F_4, F_7
C_4	Comfort	$F_{16}, F_{11}, F_{10}, F_8$
C_5	Status	$F_{17}, F_{14}, F_{15}, F_{13}, F_{12}, F_6$

Figure 1 shows the average membership degree across all categories for both studies. The reader may observe that the maximal average membership degree corresponds to the category C_2 in both scenarios. This result is certainly interesting but unsurprising because Belgian people regularly have well-paid jobs, and thus they are more focused on finding jobs with safer contract terms. Therefore, the membership degrees for the "Stability" category will be used to solve the weighted aggregation problem.

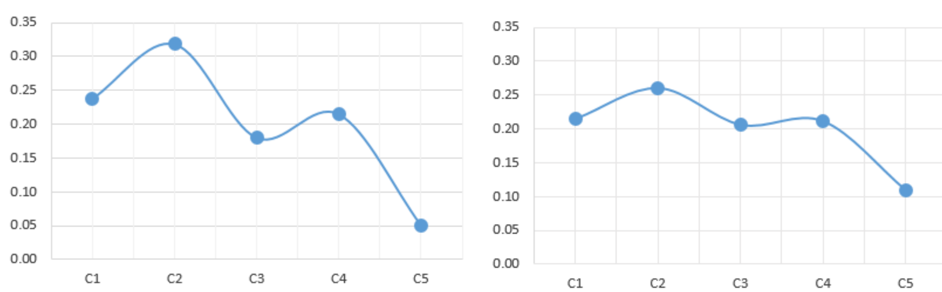

Figure 1: Average memberships degrees across predefined categories: (a) top-K aggregation scenario, and (b) top-K_i aggregation scenario.

In the first scenario, the differences are more evident given the lower dispersion in the selected factors (i.e., the respondents tend to include similar factors into the top-5). On the other hand, in the second scenario, the dispersion increases since each respondent could select a large number of relevant factors. It should be highlighted that solving the weighted Kemeny ranking problem for scenarios with high dispersion is undoubtedly more complicated due to the lack of consensus among respondents.

Figure 2 shows the performance of the selected algorithms for the top-5 aggregation scenario. The performance measure refers to normalized Hausdorff dissimilarity between the aggregated ranking and the partial rankings. Due to the stochastic nature of evolutionary and swarm intelligence algorithms, each record is computed from the average of 10 independent trials. The reader may observe that all optimizers are capable of outperforming the baseline methods, while ACS stands as the best-performing algorithm followed by MMAS. Moreover, the results have shown suggest that OX2-ISM is the best-performing GA-based optimizer, being ranked third overall.

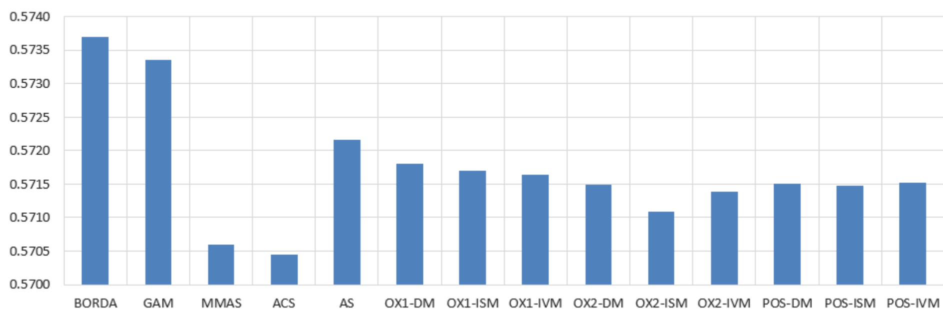

Figure 2: Performance of selected algorithms for the top-5 aggregation scenario.

Figure 3 reports the relative improvement rate of swarm and evolutionary algorithms with regards to the two greedy methods under consideration. In this aggregation scenario, the ACS and MMAS algorithms exhibit the largest improvement rates.

Figure 4 depicts the normalized Hausdorff dissimilarity measure for the top-K_i aggregation scenario. In this case, all ACO-based algorithms outperform the other approaches and ACS emerges as the top contender. The evolutionary optimizers perform comparably among them, although algorithms using the OX1 crossover operator fare slightly worst. It should be highlighted that the solutions computed by GAM are better than those produced by the weighted Borda counting method. However, this greedy approach is not competitive against evolutionary and swarm intelligence optimizers.

Figure 5 illustrates the relative improvement rate of swarm and evolutionary algorithms in comparison to the BORDA and GAM baseline methods. Observe that all optimizers are capable of outperforming the baseline meth-

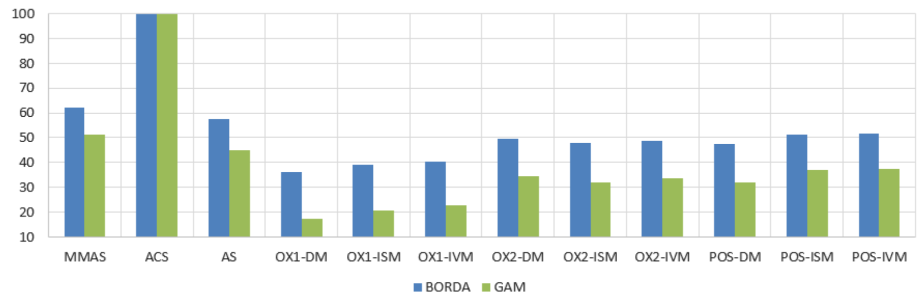

Figure 3: Improvement rate attained by the algorithms under discussion with regards to the two baseline methods for the top-K aggregation scenario.

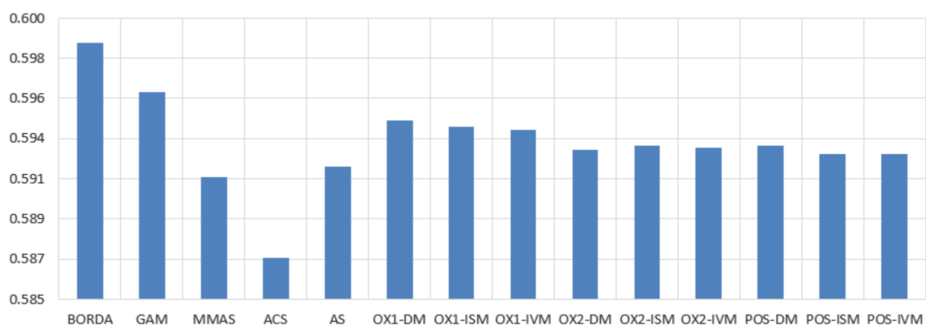

Figure 4: Performance of the selected algorithms for the top-K_i aggregation scenario.

ods and that ACS achieves the largest improvement rates.

The top-K_i datasets reveal a greater dispersion around the global (often unknown) consensus since there are fewer tied factors and finding the Kemeny ranking solution could be more challenging. Nevertheless, the inclusion of the membership degrees of each ranking to the dominant category will probably reduce the dispersion degree. This suggests that partial rankings with high membership degree to the C_2 category will comprise similar relevant factors and therefore the search problem will likely be easier to solve.

Overall, the results support the superiority of ACS and MMAS over the two greedy and nine evolutionary algorithms. The AS scheme is less competitive, which suggests that exploiting only the best solutions found by artificial ants could be convenient when aggregating partial rankings. In the next subsection, we evaluate our methodology using more generic datasets.

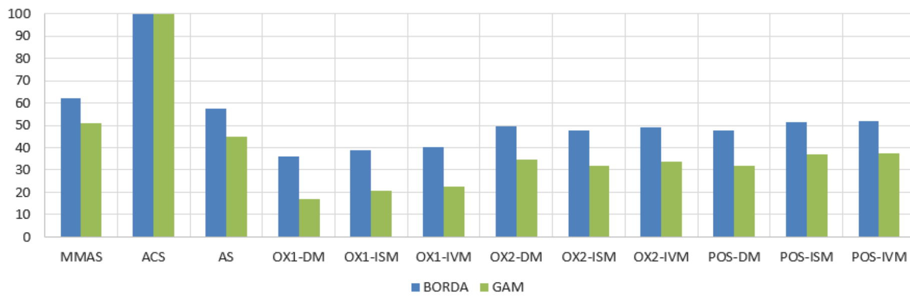

Figure 5: Improvement rate attained by the benchmarking algorithms over the two baseline methods for the top-K_i aggregation scenario.

7.3. Numerical simulations for synthetic datasets

Aiming at generalizing the above results, we adopted 190 synthetic datasets from `www.PrefLib.org` having different complexity in both the number of instances and factors. These datasets comprise top-K_i partial rankings where ties are allowed. The number of instances ranges from 10 to 10,335 whereas the number of factors goes from 4 to 155. The reader may observe that such datasets do not include an explicit definition of categories. Therefore, in these synthetic aggregation problems, the weight of each partial ranking is calculated according to the ratio of non-tied items as explained in section 6.

In order to verify the existence of significant differences among the suite of benchmarking algorithms, we computed the Friedman two-way analysis of variances by ranks [51]. The Friedman test is a multiple-comparisons nonparametric statistical test that detects whether at least two of the samples (in a set of $N > 2$ samples) represent populations with different median values or not. In our case, the Friedman test suggests rejecting the null hypothesis H_0 (p-value $= 2.2825\text{E-}10 < 0.05$) using a confidence interval of 95%. Thus, we can conclude that there are statistically significant differences between at least two algorithms across all datasets.

Figure 6 portrays the mean ranks computed by the Friedman test. From such results, we can formalize some empirical conclusions about the performance of the methods under consideration:

- The ACS algorithm is capable of notably outperforming the remaining search methods, followed by MMAS. However, the existence of statistically significant differences between them must be verified.

- The OX1 crossover operator leads to poor performance and may not be adequate for solving the extended Kemeny ranking problem. The other evolutionary optimizers perform comparably among them, with the OX2-ISM scheme producing better results.

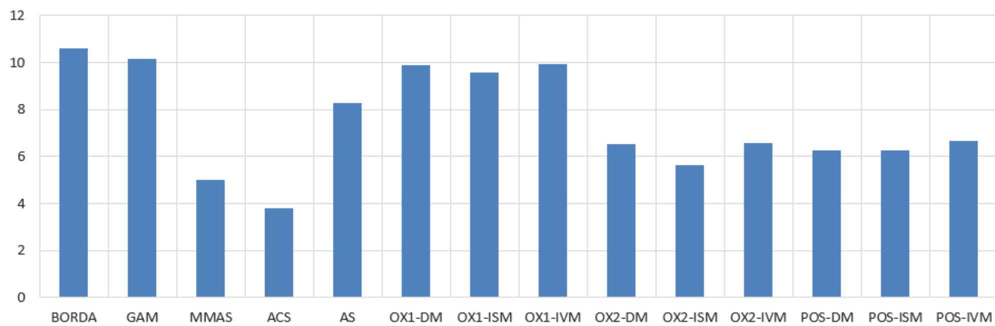

Figure 6: Mean ranks computed by the Friedman test for each algorithm across all synthetic datasets.

To further confirm the superiority of the search methods over the greedy methods, Figure 7 displays the improvement rate attained by swarm and evolutionary algorithms over these two approaches. For these synthetic datasets, ACS displays the largest improvement rates, while AS and the evolutionary methods based on the OX1 crossover operator report the worst ones.

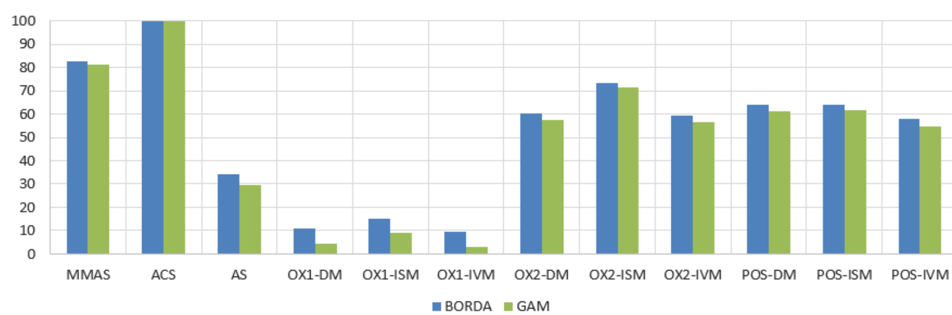

Figure 7: Improvement rate attained by the algorithms under discussion with regards to the two baseline methods for the synthetic datasets.

The last experiment is focused on determining whether the superiority of the ACS method is statistically significant or not. With this goal in mind, we use four post-hoc procedures (i.e., Bonferroni, Holm, Hochberg and Hommel)

[52] for multiple pairwise comparisons and a control method. These statistical procedures are required since in pairwise analysis, if we try to draw a conclusion involving more than one pairwise comparison, we will accumulate an error coming from its combination. Therefore, we are losing control on the Family-Wise Error Rate (FWER), defined as the probability of making one or more false discoveries among all the hypotheses when performing multiple pairwise tests [52]. Table 5 reports the unadjusted p-value and the adjusted p-values associated with each pairwise comparison using the best-performing algorithm (ACS) as the control method.

Table 5: Post-hoc procedures for pairwise comparisons using ACS as the control algorithm.

Algorithm	p-value	Bonferroni	Holm	Hochberg	Hommel
BORDA	1.8939E-56	2.4621E-55	2.4621E-55	2.4621E-55	2.4621E-55
GAM	2.0719E-49	2.6935E-48	2.4863E-48	2.4863E-48	2.4863E-48
OX1-IVM	1.8769E-46	2.4399E-45	2.0645E-45	2.0645E-45	2.0645E-45
OX1-DM	3.0934E-45	4.0214E-44	3.0934E-44	3.0934E-44	3.0934E-44
OX1-ISM	4.9068E-41	6.3789E-40	4.4161E-40	4.4161E-40	4.4161E-40
AS	2.0687E-25	2.6893E-24	1.6550E-24	1.6550E-24	1.6550E-24
POS-IVM	2.5414E-11	3.3038E-10	1.7789E-10	1.7789E-10	1.7789E-10
OX2-IVM	1.2108E-10	1.5741E-9	7.2652E-10	7.2652E-10	7.2652E-10
OX2-DM	2.8047E-10	3.6461E-9	1.4023E-9	1.4023E-9	1.4023E-9
POS-DM	1.1409E-8	1.4831E-7	4.5636E-8	3.9519E-8	3.4227E-8
POS-ISM	1.3173E-8	1.7125E-7	4.5636E-8	3.9519E-8	3.9519E-8
OX2-ISM	2.2062E-5	2.8681E-4	4.4125E-5	4.4125E-5	4.4125E-5
MMAS	0.0055	0.0725	0.0055	0.0055	0.0055

All post-hoc procedures reject the null hypothesis for a 5% significance level (corresponding to the 95% confidence interval). Only the Bonferroni procedure accepts the conservative hypothesis for the ACS-MMAS pair. This allows claiming the statistical superiority of the ACS optimizer when tested with the synthetic datasets used for simulations. From these results we can state that the strong elitism embedded into ACS/MMAS is a determinant aspect when aggregating weighted partial rankings.

8. Concluding remarks

In this paper we addressed the weighted aggregation of top-K and top-K_i partial rankings by extending the Kemeny ranking problem. To do that, we proposed a search method rooted on Ant Colony Optimization that automatically estimates the heuristic information from the rank population. Moreover, we discussed two strategies to compute the weight of each partial ranking. In the first case, the weight is computed from the fuzzy membership degree of the target instance to a set of predefined categories (i.e., information granules resulting from a partition of the whole item set). If such categories are not available, then the weight is calculated as the ratio of non-tied items. This heuristic assumes that partial rankings having fewer number of tied factors are more informative when aggregating partial rankings. The reader may observe that other alternatives to determine the weights could be explored.

During the simulations, we compared the performance of our approach against two (greedy) baseline methods and nine genetic-algorithm-based implementations in presence of both a real-world problem and 190 synthetic datasets. The results have shown that the ACS algorithm is capable of significantly outperforming the remaining techniques, followed by the MMAS algorithm. This finding could be ascribed to the elitist approach of ACS and MMAS in conjunction with the proposed strategy for estimating the heuristic information. Moreover, we observed that the OX1 crossover operator reports poor performance; therefore, it may not be suitable to solve the Kemeny ranking problem. As a future work, we will focus on extending the proposed approach to more generic aggregation scenarios.

Acknowledgement

The authors would like to thank PhD Student Isel Grau from Vrije Universiteit Brussel, Belgium, for her valuable support on designing the membership measure and running the algorithms. This work was supported by the Research Council of Hasselt University.

References

[1] J. C. Carbajal, A. McLennan, R. Tourky, Truthful implementation and preference aggregation in restricted domains, Journal of Economic Theory 148 (3) (2013) 1074–1101.

[2] S. Dhamal, Y. Narahari, Scalable preference aggregation in social networks, in: First AAAI Conference on Human Computation and Crowdsourcing, 2013.

[3] C.-L. Hwang, M.-J. Lin, Group decision making under multiple criteria: methods and applications, Vol. 281, Springer Science & Business Media, 2012.

[4] S. E. Lacy, M. A. Lones, S. L. Smith, A comparison of evolved linear and non-linear ensemble vote aggregators, in: Evolutionary Computation (CEC), 2015 IEEE Congress on, IEEE, 2015, pp. 758–763.

[5] Q.-Z. Huang, Consensus analysis of multi-agent discrete-time systems, Acta Automatica Sinica 38 (7) (2012) 1127–1133.

[6] Y. Ronin, D. Mester, D. Minkov, R. Belotserkovski, B. Jackson, P. Schnable, S. Aluru, A. Korol, Two-phase analysis in consensus genetic mapping, G3: Genes— Genomes— Genetics 2 (5) (2012) 537–549.

[7] J. C. de Borda, Mémoire sur les élections au scrutin.

[8] N. De Condorcet, et al., Essai sur l'application de l'analyse à la probabilité des décisions rendues à la pluralité des voix, Cambridge University Press, 2014.

[9] A. Rajkumar, S. Agarwal, A statistical convergence perspective of algorithms for rank aggregation from pairwise data, in: Proceedings of the 31st International Conference on Machine Learning, 2014, pp. 118–126.

[10] R. C. Prati, Combining feature ranking algorithms through rank aggregation, in: Neural Networks (IJCNN), The 2012 International Joint Conference on, IEEE, 2012, pp. 1–8.

[11] K. J. Arrow, Social choice and individual values, Vol. 12, Yale University Press, 2012.

[12] J. G. Kemeny, J. L. Snell, Mathematical models in the social sciences, Ginn and Company, 1962.

[13] V. Conitzer, A. Davenport, J. Kalagnanam, Improved bounds for computing kemeny rankings, in: AAAI, Vol. 6, 2006, pp. 620–626.

[14] A. Ali, M. Meilă, Experiments with Kemeny ranking: What works when?, Mathematical Social Sciences 64 (1) (2012) 28–40.

[15] J. A. Aledo, J. A. Gámez, D. Molina, Tackling the rank aggregation problem with evolutionary algorithms, Applied Mathematics and Computation 222 (2013) 632–644.

[16] M. Dorigo, G. D. Caro, L. M. Gambardella, Ant algorithms for discrete optimization, Artificial life 5 (2) (1999) 137–172.

[17] A. P. Engelbrecht, Fundamentals of computational swarm intelligence, John Wiley & Sons, 2006.

[18] V. Conitzer, T. Sandholm, Common voting rules as maximum likelihood estimators, arXiv preprint arXiv:1207.1368.

[19] J. González-Pachón, C. Romero, Aggregation of partial ordinal rankings: an interval goal programming approach, Computers & Operations Research 28 (8) (2001) 827–834.

[20] A. Klementiev, D. Roth, K. Small, I. Titov, Unsupervised rank aggregation with domain-specific expertise, Urbana 51 (2009) 61801.

[21] A. Klementiev, D. Roth, K. Small, Unsupervised rank aggregation with distance-based models, in: Proceedings of the 25th international conference on Machine learning, ACM, 2008, pp. 472–479.

[22] A. Ammar, D. Shah, Efficient rank aggregation using partial data, in: ACM SIGMETRICS Performance Evaluation Review, Vol. 40, ACM, 2012, pp. 355–366.

[23] S. Negahban, S. Oh, D. Shah, Iterative ranking from pair-wise comparisons, in: Advances in Neural Information Processing Systems, 2012, pp. 2474–2482.

[24] D. McFadden, et al., Conditional logit analysis of qualitative choice behavior.

[25] F. J. Brandenburg, A. Gleißner, A. Hofmeier, Comparing and aggregating partial orders with Kendall Tau distances, Discrete Mathematics, Algorithms and Applications 5 (02) (2013) 1360003.

[26] H. Young, A. Levenglick, A consistent extension of condorcet's election principle, SIAM Journal on Applied Mathematics 35 (1978) 285–300.

[27] G. Nápoles, Z. Dikopoulou, E. Papageorgiou, R. Bello, K. Vanhoof, Prototypes construction from partial rankings to characterize the attractiveness of companies in belgium, Applied Soft Computing (2016) (In Press).

[28] M. G. Kendall, A new measure of rank correlation, Biometrika 30 (1/2) (1938) 81–93.

[29] H. F, Grundzge der Mengenlehre, Leipzig, 1914.

[30] C. Spearman, footrulefor measuring correlation, British Journal of Psychology, 1904-1920 2 (1) (1906) 89–108.

[31] L. Goddman, W. Kruskal, Measures of association for cross classification, Journal of the American Statistical Association 49 (1954) 732–764.

[32] R. Fagin, R. Kumar, M. Mahdian, D. Sivakumar, E. Vee, Comparing partial rankings, SIAM Journal on Discrete Mathematics 20 (3) (2006) 628–648.

[33] P. Diaconis, R. L. Graham, Spearman's footrule as a measure of disarray, Journal of the Royal Statistical Society. Series B (Methodological) (1977) 262–268.

[34] G. Nápoles, Z. Dikopoulou, E. Papageorgiou, R. Bello, K. Vanhoof, Aggregation of partial rankings-an approach based on the kemeny ranking problem, in: Advances in Computational Intelligence, Springer, 2015, pp. 343–355.

[35] M. Dorigo, E. Bonabeau, G. Theraulaz, Ant algorithms and stigmergy, Future Generation Computer Systems 16 (8) (2000) 851–871.

[36] R. Falcon, X. Li, A. Nayak, I. Stojmenovic, The one-commodity traveling salesman problem with selective pickup and delivery: An ant colony approach, in: Evolutionary Computation (CEC), 2010 IEEE Congress on, IEEE, 2010, pp. 4326–4333.

[37] S. Tabakhi, A. Najafi, R. Ranjbar, P. Moradi, Gene selection for microarray data classification using a novel ant colony optimization, Neurocomputing 168 (2015) 1024–1036.

[38] X. Zhang, W. Chen, B. Wang, X. Chen, Intelligent fault diagnosis of rotating machinery using support vector machine with ant colony algorithm for synchronous feature selection and parameter optimization, Neurocomputing 167 (2015) 260–279.

[39] M. Dorigo, V. Maniezzo, A. Colorni, Ant system: optimization by a colony of cooperating agents, Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on 26 (1) (1996) 29–41.

[40] M. Dorigo, L. M. Gambardella, Ant colony system: a cooperative learning approach to the traveling salesman problem, Evolutionary Computation, IEEE Transactions on 1 (1) (1997) 53–66.

[41] T. Stützle, H. H. Hoos, Max–min ant system, Future generation computer systems 16 (8) (2000) 889–914.

[42] K. Socha, J. Knowles, M. Sampels, A max-min ant system for the university course timetabling problem, in: Ant algorithms, Springer, 2002, pp. 1–13.

[43] G. Gutin, A. P. Punnen, The traveling salesman problem and its variations, Vol. 12, Springer Science & Business Media, 2006.

[44] P. Larrañaga, C. M. H. Kuijpers, R. H. Murga, I. Inza, S. Dizdarevic, Genetic algorithms for the travelling salesman problem: A review of representations and operators, Artificial Intelligence Review 13 (2) (1999) 129–170.

[45] G. Syswerda, Schedule optimization using genetic algorithms, Handbook of genetic algorithms.

[46] L. Davis, Applying adaptive algorithms to epistatic domains., in: IJCAI, Vol. 85, 1985, pp. 162–164.

[47] D. B. Fogel, An evolutionary approach to the traveling salesman problem, Biological Cybernetics 60 (2) (1988) 139–144.

[48] Z. Michalewicz, Genetic algorithms + data structures = evolution programs, Springer Science & Business Media, 2013.

[49] D. B. Fogel, Applying evolutionary programming to selected traveling salesman problems, Cybernetics and systems 24 (1) (1993) 27–36.

[50] Z. Dikopoulou, G. Nápoles, E. Papageorgiou, K. Vanhoof, Multi criteria methods used for assessing companies' attractiveness, in: Multiple Criteria Decision Making (MCDM 2015), International Conference on, 2015.

[51] M. Friedman, The use of ranks to avoid the assumption of normality implicit in the analysis of variance, Journal of the American Statistical Association 32 (200) (1937) 675–701.

[52] J. Luengo, S. García, F. Herrera, A study on the use of statistical tests for experimentation with neural networks: Analysis of parametric test conditions and non-parametric tests, Expert Systems with Applications 36 (4) (2009) 7798–7808.