

Clustering Decision Makers with respect to similarity of views

Edward Abel^a, Ludmil Mikhailov^b, John Keane^a

^a School of Computer Science
The University of Manchester
UK, M13 9PL

^b Manchester Business School
The University of Manchester
UK, M15 6PB

Abstract—Within a large group of decision makers, varying amounts of both conflicting and harmonious views will be prevalent within the group, but obscured due to group size. When the number of Decision Makers is large, utilizing clustering during the process of aggregation of their views should aid both knowledge discovery - about the group's conflict and consensus - as well as helping to streamline the aggregation process to reach a group consensus. We conjecture that this can be realized by using the similarity of views of a large group of decision makers to define clusters of analogous opinions. From each cluster of decision makers, a representation of the views of its members can then be sought. This set of representations can then be utilized for aggregation to help reach a final whole group consensus.

Keywords—Multi-criteria decision making, Pairwise comparison, Inconsistency, Multi-objective optimization, Genetic algorithms, Clustering.

I. INTRODUCTION

Multi-Criteria Decision Making (MCDM) seeks to determine the suitability of alternative outcomes of a decision with respect to a number of criteria. The notion of Pairwise Comparison (PC) is employed extensively by MCDM methods such as AHP [1] as well as employable within methods such as TOPSIS [2] to aid the calculation of the weights of importance of the criteria. PCs help to facilitate a separation of concerns through allowing a Decision Maker (DM) to consider only a pair of decision elements and to determine their preference between the pair. From a set of PCs, one for each pairing of elements in a set of decision elements, a one-dimensional *weights vector* can be derived that represents a ranking of the set of elements under consideration by the DM. For many real world decisions the opinions of multiple DMs is utilized, to either benefit from their combined expertise or to incorporate conflicting views and experiences. When utilized within a group environment, the process of deriving a weights vector from multiple DMs opinions needs to incorporate the aggregation of the group of DMs' PCs into the formulation of a single weights vector for the group.

When the number of DMs is large, we propose utilising clustering during the aggregation process, to group together those DMs whose views are similar and then look to aggregate the views of these sub-groups. This should aid

both knowledge discovery regarding the group's conflict and consensus, as well as helping to streamline the aggregation process to reach a group consensus.

Additionally DMs are subject to irrationalities such as inconsistencies which can adversely affect decision outcomes. Inconsistency within a set of PCs is the extent to which the set of judgments are coherent. When inconsistency is present it can have adverse effects upon the accuracy of any derived weights vector [3]. Therefore, seeking to reduce inconsistency during the aggregation process should help reduce its adverse effects.

This paper presents an approach to the aggregation of PC judgments of a large group of DMs, whilst simultaneously seeking to reduce inconsistency. The approach first utilizes clustering - to group DMs with similar views into subgroups, then employs Multi-Objective Optimization (MOO) - to model compromise between each subgroup of DMs with respect to their views to find aggregated whole group solutions. From the clustering stage we can glean knowledge about the makeup of the group to help to determine for example, if the group is a homogeneous unit or a collection of defined subgroups, which can impact aggregation [4]. Measures of evaluation of the clustering have been defined to aid consideration of the number of clusters in which to group the DMs. The MOO stage then seeks to optimize multiple objectives simultaneously to find a set of trade-off solutions between the conflicting views of each subgroup of DMs. For these *non-dominated solutions*, improvement in any objective of the problem will result in a decrease within one or more of the other objectives. Together they map out the trade-off front of the problem. A range of measures have been defined that can be utilized as objectives to measure compromise between a set of judgments and an aggregated set of judgments. Constraints can also be utilized within the approach by each cluster of DMs, to represent the tolerance of compromise regarding the amount of concession they will tolerate in the pursuit of finding aggregated group solutions. From the trade-off front of solutions found during the MOO stage a single solution from the set of trade-off solutions can be selected, when required, based on utilizing knowledge of the total level of compromise of the group to reach consensus.

The rest of the paper is structured as follows: Section II defines and discusses the problem of group aggregation of PCs, along with discussion of clustering and inconsistency reduction; our approach to the aggregation of a large group of DMs is then outlined and defined in Section III; Section IV discusses examples of our approach; conclusions are given in Section V.

II. AGGREGATION OF JUDGMENTS OF A GROUP OF DMs

The process of aggregation of multiple DMs' PCs is examined in more detail and previous approaches discussed. This is followed by discussions of inconsistency reduction and clustering.

A. Aggregation of PC judgments of a group of DMs

PC allows a DM to consider only a pair of decision elements and to determine their preference, and strength of preference, between the pair, with respect to an intangible factor. This division of a larger decision problem can be achieved through the use of the Law of Comparative Judgment [5]. Given two elements x and y , we can denote that a DM prefers element x to element y with the notation $x \succ y$. Various numerical scales may be utilized to represent the strength of preference; the most prevalent being the Saaty 1-9 scale [6], where, for example, if element x is preferred 3 times more than element y , this can be denoted as $x \succ y$ with a preference strength of 3. If neither element is preferred over the other, the elements are said to be equally preferred, denoted by $x \sim y$ and represented with the value 1. The set of PCs, one for each pairing of elements in a set of elements, along with the self-comparison values and the reciprocal values, can be collated into a two-dimensional Pairwise Comparison Matrix (PCM), as shown in (1) for a set of n elements, where a_{ij} represents the judgment between elements i and j .

$$PCM = \begin{pmatrix} 1 & a_{12} & \cdots & a_{1n} \\ 1/a_{12} & 1 & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 1/a_{1n} & 1/a_{2n} & \cdots & 1 \end{pmatrix} \quad (1)$$

For a completed PCM of the type (1) of $(n \times n)$ elements, there exists a weights vector $w = \{w_1, w_2, \dots, w_n\}^T$, where w_i represents the weighting of the element i for $i=1$ to n . A weights vector can be derived through the use of a Prioritization Method (PM). Many PMs exist for this task; see [7] for a comprehensive discussion of PMs. PCs can similarly be utilized when multiple DMs provide their preferences to create multiple PCMs. In this case a single weights vector, representing the combined preferences of all the DMs is to be derived. Given a set of D DMs giving their PC preferences for a set of n elements, the problem is to aggregate the PCM's of the DMs to derive a single weights vector. This may be achieved through Aggregation of

Individual Judgments (AIJ) - deriving an aggregated PCM from which a single weights vector can then be derived, or via Aggregation of Individual Priorities (AIP) - calculation of a separate weights vector for each DM, from which a single weights vector can be calculated through the aggregation of the set of weight vectors. Our approach deals with AIJ as it allows inconsistency to be considered and reduced within the aggregation process, before weights vector priorities are derived.

The Geometric Mean Method (GMM) [8] can be used to aggregate the PCMs of multiple DMs into a single aggregated PCM. GMM looks to find an aggregated PCM without consideration of the compromise needed between each DM to reach consensus and no levels of tolerance to compromise can be set. The Arithmetic Mean Method (AMM) [9] can also be utilized to calculate an aggregated PCM of judgments using the arithmetic mean for the synthesis calculation. As with the GMM, the AMM does not consider the levels of compromise upon each DM's judgments. A method proposed in [10] considers the level of compromise upon each DM's judgments during aggregation for a small group. Building on this, we now propose an approach for when the number of DM's is large, utilizing clustering to first group DMs into congruent subgroups before aggregating based upon representational judgments of each subgroups member's.

B. Inconsistency

The consistency of a PCM is the extent to which its judgments are coherent. When there is inconsistency present in a PCM, any weights vector derived from it will only be an estimate of its true preferences. Consequently, different PMs may then derive different weights vector estimates. Inconsistency within a PCM of more than a handful of elements has been shown to be almost inevitable [7] and therefore needs to be considered. Inconsistency may be ordinal or cardinal in nature.

Ordinal inconsistency identifies inconsistent information independently of consideration of the strengths of preference of a DM's judgments. We denote that an element x is preferred to another element y with the notation $x \succ y$. Given a set of 3 elements, a , b and c : if $a \succ b$, $b \succ c$ and $c \succ a$, then the judgments are intransitive and ordinal inconsistency is present in the form of a 3-way-cycle. The total number of 3-way cycles present can be used as a measure of ordinal inconsistency in a PCM. The presence of 3-way cycles can be determined via an algorithm proposed in [11]. This can also be utilized to determine the total number of 3-way cycles within a PCM, usually denoted as L . Cardinal inconsistency identifies inconsistency between a set of judgments taking into account the strength of preference of the judgments. The Consistency Ratio (CR), proposed by Saaty [6], can be utilized to measure the level of cardinal inconsistency of a PCM. The lower the CR value, the lower the amount of cardinal inconsistency present in the PCM. Saaty further proposed an acceptability threshold value of a PCM's CR value [6]. The threshold is designed to be an indicator as to whether a PCM is consistent enough for a

satisfactory weights vector estimate to be derived. Using this threshold, a PCM with a CR value of 0.1 or less is considered to be acceptable.

Previous approaches have looked to reduce inconsistency in a single PCM as a separate process (to that of group aggregation) to find an altered PCM with reduced inconsistency. They are generally approaches that consider either ordinal or cardinal inconsistency only and look to converge to a single predetermined fixed value of inconsistency, see [12], [13].

C. Clustering

We can look to group DMs with similar views through clustering. Clustering discovers natural groupings of a set of points or objects. [14]. Such clustering can be performed by the K-means clustering approach. Given a set of d dimensional instances, K-means seeks to cluster the instances into a set of K clusters, such that the squared error between the mean point of a cluster and its points are minimised. [15].

The K-means algorithm has three stages:

1. The instances are assigned randomly into k clusters;
2. For each cluster the centroids between its members is calculated and then the distance each instance is from each cluster centroid is determined;
3. Each instance is then assigned to the cluster with the nearest centroid.

Stages 2 and 3 are then repeated until no instances are assigned to a new cluster in Stage 3.

A limitation of K-means is that as the instances are initially assigned randomly to clusters it is possible that a sub optimal convergence is reached. The K-means++ algorithm [16] is an enhancement to K-means that seeks to reduce this limitation through a modified first stage that aims to initially assign the instances into clusters such that the initial clustering is closer to an optimal initialisation, see [16]. The k-means++ algorithm is utilized within our approach.

III. OUR APPROACH

When we have a large number of DMs our approach seeks to cluster them into subgroups based upon the similarity of their views. Therefore DMs with analogous views will be in the same cluster and DMs with conflicting views in different clusters. We then derive a single aggregated PCM representation for each subgroup. We utilise these aggregated PCMs as part of a MOO approach to find aggregated solutions between the conflicting subgroups. Clustering the DMs into sub-groups will reduce the number of objectives needed as part of the MOO approach to reach aggregation. Additionally, via clustering the DMs on similarity of their views, it also seeks to reduce the amount of redundancy in the MOO stage as DMs with similar views will be grouped as one. The stages of our approach are shown in Fig 1.

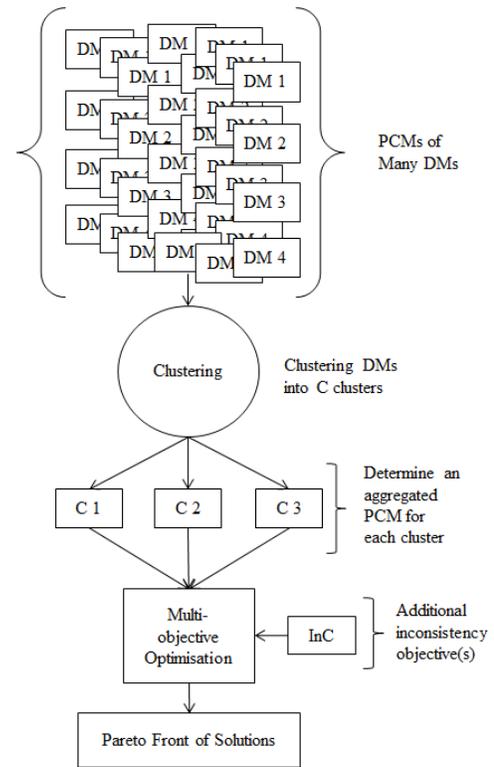


Fig 1: Aggregation Process

The approach has three stages:

1. The set of DMs are clustered into subgroups based upon the similarity of their ranking of the set of elements;
2. From each cluster a single aggregated PCM is derived as representative of the subgroup;
3. The derived Aggregated PCM of each cluster is utilized within a MOO approach to find non-dominated aggregated solutions between sub-groups.

1) Clustering the DMs

The DMs are grouped into clusters based upon the similarity of their views regarding their ranking of the elements of the problem. Given a problem with n elements and D decision makers who each define a complete n by n PCM of their judgments $\{PCM_1, PCM_2, \dots, PCM_D\}$, for each DM we derive a weights vector of their judgments of the n elements. In the examples presented, the Geometric Mean PM [17] is utilized to derive weights vectors, however any PM can be utilized within our approach. This set of D weights vectors is then utilized as the feature vector input for the clustering stage. This way DMs with similar ranking of the elements will be grouped together into clusters. The set of D DMs are clustered into C clusters using k-means++. The feature set of 5 DMs' initial PCMs and their rankings of a set of 4 elements is shown in Fig 2.

2) Deriving Aggregated PCMs for Clusters

From each cluster a single aggregated PCM is derived from its members utilizing a single objective optimisation

process. For each DM's PCM the minimum number of judgments (J) needed to contain all the information within each PCM can be represented as:

$$J = n(n-1)/2 \quad (2)$$

Thus, a Judgment Set O of cardinality J can be selected, containing enough information to reconstruct the whole of the PCM for that DM. O can be represented as the upper triangle of a PCM. We can model an O representation of each DM's PCM $\{O_1, O_2, \dots, O_D\}$, each of which consists of J judgments $\{o_1^k, o_2^k, \dots, o_J^k\}$ for $k=1, \dots, D$. Each cluster's aggregated PCM is derived via a single objective optimisation process looking to optimise a total measure of compromise (see section C below) between the judgments of its members and the judgments of an aggregated PCM. In this way an Aggregated PCM is sought for each cluster that is most representational of the cluster's members through minimising the total compromise to its member's views. The set of C derived Aggregated PCMs are then the input for the MOO aggregation stage. The clusters found and their aggregated PCMs can be evaluated (see section B below) with respect to how closely matching the DMs views are in each cluster, helping to determine an appropriate C parameter value.

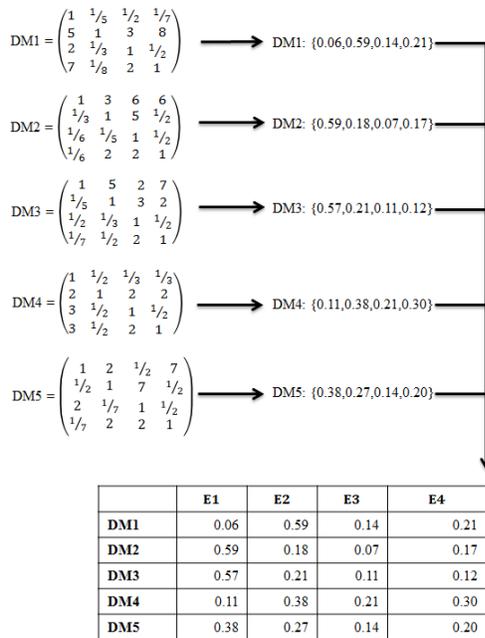


Fig 2: Clustering Feature Vector

3) MOO Aggregation

With the DMs clustered into C clusters, the Aggregated PCM representing each cluster's members are utilized within the MOO stage to find a front of non-dominated solutions for the whole group. Given the set of C aggregated PCMs, one for each cluster $\{CPCM_1, CPCM_2, \dots, CPCM_C\}$, we

can represent each cluster's Aggregated PCM as judgment sets $\{O_1, O_2, \dots, O_C\}$, again each consisting of J judgments $\{o_1^k, o_2^k, \dots, o_J^k\}$ for $k=1, \dots, C$. We seek the set of non-dominated Aggregated Consensus Solutions, which we can represent as a judgment set of cardinality J , denoted as $A = \{a_1, a_2, \dots, a_j\}$. The set A represents the decision variables that can be obtained by minimizing the set of objectives. We seek to minimise the set of objectives consisting of two subsets. The first subset representing measures of compromise objectives of cardinality C , that each seeks to minimise the measure of compromise with respect to the corresponding O of each cluster. The second subset represents any additional inconsistency objectives. The approach additionally allows constraints to be set on the objectives, both on measures of compromise objectives and upon inconsistency objectives. Our approach then seeks to simultaneously optimise this set of objectives to find the trade-off front of non-dominated PCM solutions to the problem. Knowledge pertaining to the levels of conflict between the clusters can be gleaned from examination of the trade-off front. A weights vector representing the aggregated solution for the whole group can then be derived from any of the non-dominated PCM solutions found. For example if the 5 DMs shown in Fig 2 are clustered into 2 clusters - the first containing DMs 1,4 and the second containing DM 2,3,5 - derived PCMs for the two clusters along with an illustration of a set of trade-off non-dominated PCM solutions are shown in Fig 3. Any solution from the trade-off front of solutions can then be selected as shown, from which a final aggregated weights vector can be derived.

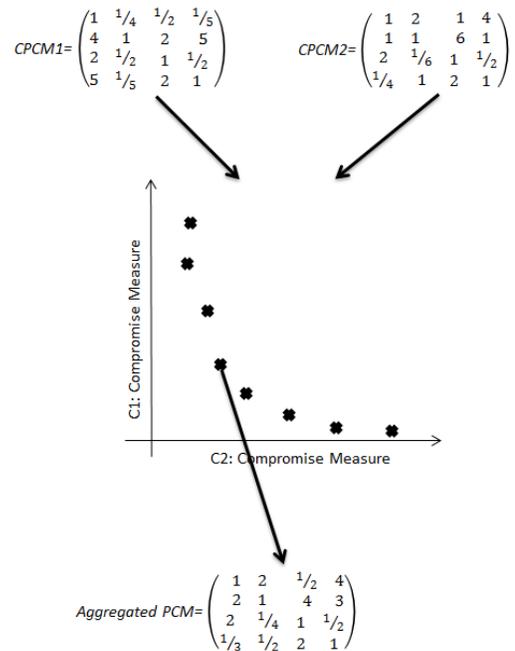


Fig 3: Aggregation Process

Finally, as required, we can additionally determine a single aggregated solution automatically from the set of non-dominated solutions found utilising a total measure of compromise such as that shown in (9). For each non-dominated solutions found, a measure of the total (global) compromise measure can be calculated. From this, a ranking of the non-dominated solutions can be sought with respect to their global measure of compromise, from which the solution with the lowest total measure of compromise can be selected and a weights vector derived. When multiple non-dominated solutions share the lowest total measure of compromise value, a single weights vector can be derived as the average (utilizing the geometric mean) of the separate weight vectors derived from this sub-set of non-dominated solutions that share the lowest value.

B. Measures of Clustering Evaluation

We can evaluate the clustering of the DMs to aid in the selection of an appropriate number of clusters in which to group the set of DMs. From each cluster we can measure the similarity between the rankings of the elements derived from a cluster's aggregated PCM and the cluster's member's initial rankings. Given a weights vector w_c derived from an Aggregated PCM of a cluster for an n element problem, $w_c = \{w_{c1}, w_{c2}, \dots, w_{cn}\}^T$ and the set of weights vectors of the cluster's members w_{c1} to w_{cm} each consisting of n weights where m is the number of DMs in the cluster $\{w_1^k, w_2^k, \dots, w_n^k\}$ for $k=1, \dots, m$. We can calculate measures of Total Ranking Deviation and Total Rank Reversals for the cluster.

1) Total Ranking Deviation (TRD)

The Total Ranking Deviation (TRD) is a measure of the total difference between each element in a cluster's aggregated weights vector and each member's weights vector for each element for all members of the cluster

$$TRD = \sum_{m=1}^m \sum_{n=1}^n abs|w_{cn} - w_n^m| \quad (3)$$

2) Total Rank Reversals (TRR)

The Total Rank Reversals (TRR) is a measure of the total number of preferences in an Aggregated PCM for a cluster that represents the reversed view within a cluster member's preferences, summed for all members of the cluster. We can represent a weights vector of size n as $n-1$ Preferences - a Preference Set. A Rank Reversal (RR) occurs when a ranking preference within a DM's Preference Set is reversed in the cluster's Preference Set. TRR is the sum of the RR for all members of the cluster:

$$TRD = \sum_{n=1}^n \sum_{m=1}^m RR_n^m \quad (4)$$

Consequently a TRR value of 0 represents a cluster for which all its members share the same ordinal rankings of the elements (but not necessarily the same cardinal weights values). Additionally we can divide the TRR value of a cluster by the number of DMs in the cluster to obtain an average measure of reversals per DM for the cluster.

C. Measures of Compromise

Given a DM judgment set (O) represented as a set of judgments $\{o_1, o_2, \dots, o_J\}$ of cardinality J . The amount of change between O and a second *Aggregated* judgment set (A) of judgments $\{a_1, a_2, \dots, a_J\}$ can be calculated using a measure of compromise. Each measure can be employed as an objective to measure the compromise between a cluster's representational judgments and an aggregated consensus solution.

1) Number of Judgment Violations (NJV)

NJV is a measure of the number of the original set of PC judgments that have changed, without consideration of the amount of change of each judgment; where σ evaluates to 0 or 1 for each Boolean evaluation:

$$NJV = \sum_{j=1}^J \sigma(o_j \neq a_j) \quad (5)$$

2) Total Judgment Deviation (TJD)

TJD is a measure of the total amount of change between the original judgments and an altered judgment set. It takes into consideration the amount of preference change between each judgment comparison:

$$TJD = \sum_{j=1}^J abs|o_j - a_j| \quad (6)$$

A modified version of TJD is Squared Total Judgment Deviation (STJD). Here the deviations between the corresponding judgments in both sets are squared; consequently altered judgments with a large alteration in strength will have a greater impact upon the measure's total:

$$STJD = \sum_{j=1}^J (o_j - a_j)^2 \quad (7)$$

3) Number of Judgment Reversals (NJR)

NJR is a measure of the number of judgments from the original set that have been inverted in an altered judgment set. For example, given an original judgment between elements x and y where $x \succ y$: if in an altered judgment set it is the case that $x \prec y$ then a judgment reversal has occurred. This measure also considers half reversals, which occur when a judgment of equal preference is altered to be a judgment of not equal preference or vice versa. The 1-9 scale can be used to specify equal preference, greater than

equal preference and less than equal preference, as 1, greater than 1 and less than 1 respectively:

$$NJR = \sum_{j=1}^J R_j \quad (8)$$

where

$$R_j \begin{cases} 1: o_j > 1 \text{ and } a_j < 1 \\ 1: o_j < 1 \text{ and } a_j > 1 \\ \frac{1}{2}: o_j = 1 \text{ and } a_j \neq 1 \\ \frac{1}{2}: o_j \neq 1 \text{ and } a_j = 1 \\ 0: \text{otherwise} \end{cases}$$

4) Total Measures of Compromise

To find an Aggregated PCM for each cluster a total measure of compromise is used. This represents the sum of a measure of compromise value for each DM within the cluster for a chosen measure of compromise. For example, the total number of judgment reversals (TNJR) for a cluster with m members can be calculated via:

$$TNJR = \sum_{m=1}^m \sum_{j=1}^J R_j \quad (9)$$

Total measures of compromise for the other measures of compromise can be calculated in a similar way. Total measures of compromise can additionally be utilized to help find an aggregated single solution from the front of non-dominated solutions found after the MOO stage.

D. Inconsistency Objectives

To help discovery of consistent non-dominated solutions during the MOO stage, inconsistency objectives can be incorporated into the MOO stage. Both cardinal and/or ordinal inconsistency measures can be incorporated. Ordinal inconsistency can be considered through employing the number of 3-way cycles (L) as an inconsistency reduction objective [11], looking to minimize the number of cycles within non-dominated solutions found. Cardinal inconsistency can be considered through utilizing the Consistency Ratio (CR) [6] as an inconsistency reduction objective. Constraints can also be set on any inconsistency objectives, for example, to adhere to Saaty's recommendation that a PCM's CR should be less than 0.1 [6], an upper limit constraint of 0.1 can be set upon a CR inconsistency reduction objective.

IV. ILLUSTRATIVE EXAMPLES

Examples of our approach are presented following an overview of its implementation: Example 1 illustrates the process with a small number of DMs to aid explanation; Example 2 illustrates the approach for a large number of DMs.

A. Approach Implementation

For the clustering stage the K-means++ algorithm is used to group the DMs. During the clustering the Euclidean distance was employed as the distance function and maximum iterations were set to 500.

Genetic Algorithms (GA) can be used to stochastically solve optimization problems and are employed to derive aggregated PCMs for each cluster. A Multiple Objective Genetic Algorithm (MOGA) is employed in the MOO stage to find a set of non-dominated solutions. The MOO stage is implemented utilizing the MOCeLL MOGA [18] in which the population of individuals is arranged as a two-dimensional grid and restricted mating ensures diversity is preserved for longer. An external archive is used to store the set of non-dominated solutions found and feedback is additionally employed to help stimulate convergence towards the problem's trade-off front. Objective constraints have been implemented utilizing Constrained Pareto Dominance [19] as defined as the constraint handling procedure within the Non-dominated Sorting Genetic Algorithm-II (NSGAI) [19]. The MOGA examples were executed employing the following parameter settings: population size of 100 (10x10 grid); maximum evaluations count of 25,000; archive size dynamically defined based upon the number of clusters and objectives within the problem with a feedback value of 25% of the size of the archive. Selection is performed via binary tournament (see [20] for more details) with single point crossover (with crossover probability 0.9) and bit flip mutation (with probability 0.01) employed (see [21] for discussions of crossover and mutation). For deriving representational aggregated PCMs for each cluster each single objective GA is employed with a population size of 100 and maximum evaluations count of 25000. Again selection is performed via binary tournament with single point crossover and bit flip mutation utilized.

B. Example 1

Table 1 shows the PCMs and priority rankings from 5 DMs for a set of 4 elements.

TABLE 1: EXAMPLE 1 - 5 DMs' ORIGINAL PCMs

		DM1				
		1	2	3	4	Pri.
1		1	1/5	1/2	1/7	0.06
2		5	1	3	8	0.59
3		2	1/3	1	1/2	0.14
4		7	1/8	2	1	0.21
		DM2				
		1	2	3	4	Pri.
1		1	3	6	6	0.59
2		1/3	1	5	1/2	0.18
3		1/6	1/5	1	1/2	0.07
4		1/6	2	2	1	0.17

DM3					Pri.
	1	2	3	4	
1	1	5	2	7	0.57
2	1/5	1	3	2	0.21
3	1/2	1/3	1	1/2	0.11
4	1/7	1/2	2	1	0.12

DM4					Pri.
	1	2	3	4	
1	1	1/2	1/3	1/3	0.11
2	2	1	2	2	0.38
3	3	1/2	1	1/2	0.21
4	3	1/2	2	1	0.30

DM5					Pri.
	1	2	3	4	
1	1	2	1/2	7	0.27
2	1/2	1	7	1/2	0.14
3	2	1/7	1	1/2	0.20
4	1/7	2	2	1	

Next a C value of 2 is used to cluster the group, and then Total STJD is utilized as the single objective in each cluster, to find an Aggregated PCM for each cluster. The pair of derived aggregated PCMs along with the DMs assigned to each cluster are shown in Table 2. The evaluation measures of these clusters are shown in Table 3. We see that when $C=2$ the DMs have been clustered such that aggregated PCMs weights vectors are derived without any rank reversals for any DM.

TABLE 2: EXAMPLE 1 - DERIVED AGGREGATED PCMS FOR THE 2 CLUSTERS

C1					Pri.
Size:2					
DMS: {1,4}					
	1	2	3	4	
1	1	1/4	1/2	1/5	0.08
2	4	1	2	5	0.51
3	2	1/2	1	1/2	0.17
4	5	1/5	2	1	0.24

C2					Pri.
Size:3					
DMS: {2,3,5}					
	1	2	3	4	
1	1	3	3	7	0.55
2	1/3	1	5	1	0.22
3	1/3	1/5	1	1/2	0.08
4	1/7	1	2	1	0.14

TABLE 3: EXAMPLE 1 CLUSTER EVALUATION MEASURES

	TRD	TRR
C1	0.43	0
C2	0.55	0

Next utilizing STJD as the objective measure of compromise for each cluster in the MOO stage, we derive a set of non-dominated solutions for this 2 objective set, the objective space of which is shown in Fig 4. We see there are a sub-set of the non-dominated solutions which share the lowest Total (global) STJD value. From this sub-set a single weights vector for the whole group can be derived (see Table 4) as the average (utilizing the geometric mean) of the

separate weight vectors derived from this sub-set of non-dominated solutions that share the lowest Total STJD value.

TABLE 4: EXAMPLE 1 OVERALL AGGREGATED WEIGHTS VECTOR

1	2	3	4
0.2584	0.4431	0.1286	0.1699

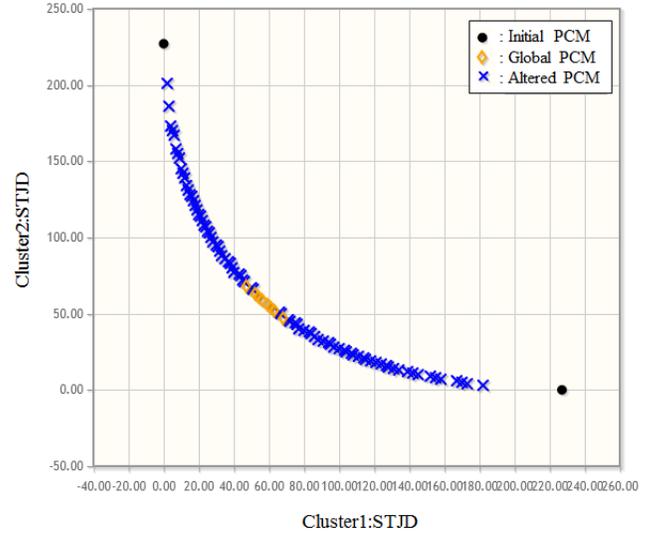


Fig 4: Example 1 Objective space

C. Example 2

Now let us consider an example with a much larger number of DMs, that of a set of 4 elements to be ranked by 100 DMs. We will look to cluster the 100 DMs into 5 clusters, then utilize the Total NJR measure to derive an aggregated PCM for each cluster. The 5 derived aggregated PCMs and priorities rankings are shown in Table 5 along with the number of DMs assigned to each cluster. The evaluation of the clustering is shown in Table 6.

TABLE 5: EXAMPLE 2 - DERIVED AGGREGATED PCMS FROM 5 CLUSTERS

C1					Pri.
Size:27					
	1	2	3	4	
1	1	1/9	1/2	1/9	0.04
2	9	1	8	1/8	0.24
3	2	1/8	1	1/7	0.06
4	9	8	7	1	0.66

C2					Pri.
Size:20					
	1	2	3	4	
1	1	1/5	1/5	1/4	0.65
2	5	1	1/5	1/2	0.16
3	5	5	1	9	3.00
4	4	2	1/9	1	4.00

C3					Pri.
Size:13					
	1	2	3	4	
1	1	8	1/8	1/4	0.13
2	1/8	1	1	1/8	0.06
3	8	1	1	1/4	0.21
4	4	8	4	1	0.60

C4 Size:15					Pri.
	1	2	3	4	
1	1	2	7	6	0.58
2	1/2	1	1	1	0.16
3	1/7	1	1	6	0.18
4	1/6	1	1/6	1	0.08

C5 Size:25					Pri.
	1	2	3	4	
1	1	1/3	1	2	0.16
2	3	1	6	9	0.63
3	1	1/6	1	3	0.15
4	1/2	1/9	1/3	1	0.06

From Table 6 we observe that for each cluster an aggregated PCM has been derived whose weights vector has an average total Rank Reversals lower than the number of DMs for each cluster. Thus we can conclude that the aggregated PCMs for the clusters are good representations of their member's views.

TABLE 6: EXAMPLE 2 CLUSTER EVALUATION MEASURES

	TRD	TRR	Size	Average RR Per DM
C1	16	21	27	0.78
C2	6.9	16	20	0.80
C3	3.2	12	13	0.92
C4	5.1	13	15	0.87
C5	11	23	25	0.92

Next we take the 5 aggregated PCMs and employ NJR as a measure of compromise objective for each along with a 6th objective of CR, with an added constraint upon it of 0.1. So we seek to derive a set of non-dominated solutions all of which will have a CR value less than Saaty's threshold recommendation. We could visualise the solutions in the objective space with respect to pairs of clusters to get further understanding of the conflict between them. This way we can see clusters that are closest to agreement and also clusters with outlying views. From the set of non-dominated solutions found we can then determine the sub-set of solutions with the lowest Total NJR value from which we can derive a final aggregated weights vector for the 100 DMs, shown in Table 7.

TABLE 7: EXAMPLE 2 OVERALL AGGREGATED WEIGHTS VECTOR

1	2	3	4
0.0886	0.3298	0.1601	0.4215

V. CONCLUSION

We have presented an approach to the aggregation of PC judgments of a large group of DMs through first clustering the DMs into subgroups based upon the similarity of their views. We then derive a single aggregated PCM representation for each subgroup. Using these aggregated PCMs and MOO we look to derive a set of non-dominated solutions between the clusters, whilst simultaneously seeking to reduce inconsistency.

Future work will investigate utilizing the cluster evaluation measures to help set thresholds of difference

between DMs to help define the number of clusters in which to group the DMs. We will also investigate employing different clustering approaches such as hierarchical clustering.

REFERENCES

- [1] Saaty TL. The Analytic Hierarchy Process, Planning, Priority Setting, Resource Allocation New York: McGraw-Hill; 1980.
- [2] Hwang CL, Yoon KP. Multiple attribute decision making. Methods and applications. A state-of-the-art survey. 1981.
- [3] Koczkodaj W, Szarek S. On distance-based inconsistency reduction algorithms for pairwise comparisons. Logic Journal of IGPL. 2010; 18(6): p. 859-869.
- [4] Forman E. , Peniwati K. Aggregating Individual Judgments and Priorities with the Analytic Hierarchy Process. European Journal of operational Research. ; 108: p. 165-169.
- [5] Thurstone L. A Law of Comparative Judgment. Psychological Review. 1927; 34(4): p. 266-270.
- [6] Saaty TL. A scaling method for priorities in hierarchical structures. Journal of Mathematical Psychology. 1977; 15(3): p. 234-281.
- [7] Choo EU, Wedley WC. A common framework for deriving preference values from pairwise comparison matrices. Computers & Operations Research. 2004; 31(6): p. 893-908.
- [8] Saaty TL. Group decision making and the AHP. In Golden BL. The analytic Hierarchy Process. Applications and Studies. Berlin: Springer; 1989. p. 59-67.
- [9] Ramanathan R, Ganesh LS. Group Preference Aggregation Methods Employed in AHP: an Evaluation and an Intrinsic Process for Deriving Members' Weightages. European Journal of Operational Research. 1994; 79: p. 249-265.
- [10] Abel E, Mikhailov L, Keane J. Group Aggregation of Pairwise Comparisons Using Multi-Objective Optimisation. unpublished.
- [11] Kwiesielewicz M, Uden EV. Inconsistent and contradictory judgements in pairwise comparison method in the AHP. Computers & Operations Research. 2004; 31: p. 713-719.
- [12] Wang C, Lui S, Pang C. Using genetic algorithm improve the consistency of fuzzy analytic hierarchy process. Soft Computing and Intelligent Systems (SCIS) and 13th International Symposium on Advanced Intelligent Systems (ISIS). 2012;; p. 977- 982.
- [13] Siraj S, Mikhailov L, Keane J. A heuristic method to rectify intransitive judgments in pairwise comparison matrices. European Journal of Operational Research. 2012; 216: p. 420-428.
- [14] Jain A. Data clustering: 50 years beyond K-means. Pattern Recognition Letters. 2010; 31(8): p. 651-666.
- [15] Lloyd S. Least Squares Quantization in PCM. IEEE. Transactions on information theory. 1982; IT-28(2).
- [16] Arthur D, VS. k-means++: the advantages of careful seeding. In SODA '07 Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms; 2007. p. 1027-1035.
- [17] Crawford GB. The geometric mean procedure for estimating the scale of a judgement matrix. Mathematical Modelling. 1987; 9(3-5): p. 327-334.
- [18] Nebro AJ, Durillo JJ, Luna F, Dorronsoro B, Alba E. A cellular genetic algorithm for multiobjective optimization. NISCO. 2006;; p. 25-36.
- [19] Deb K, Partap A, Meyarivan T. A Fast and Elitist Multiobjective Genetic Algorithm. NSGA-II. IEEE Transactions on evolutionary computing. 2002; 6(2): p. 182-197.
- [20] Goldberg DE, Deb K. A comparative analysis of selection schemes using in genetic algorithms. In Foundations of Genetic Algorithms.: Morgan Kaufmann; 1991. p. 69-93.
- [21] Haupt R, Haupt S. Practical Genetic Algorithms: John Wiley & Sons Inc.; 2004