Data clustering and imputing using a two-level multi-objective genetic algorithm (GA): A case study of maintenance cost data for tunnel fans

Yamur K. Aldouri1*, Hassan Al-Chalabi1 and Liangwei Zhang2

Abstract: This study develops a new two-level multi-objective genetic algorithm (GA) to optimise clustering to reduce and impute missing cost data for fans used in road tunnels by the Swedish Transport Administration (Trafikverket). Level 1 uses a multi-objective GA based on fuzzy c-means to cluster cost data objects based on three main indices. The first is cluster centre outliers; the second is the compactness and separation ($v_k$) of the data points and cluster centres; the third is the intensity of data points belonging to the derived clusters. Our clustering model is validated using k-means clustering. Level 2 uses a multi-objective GA to impute the reduced missing cost data in volume using a valid data period. The optimal population has a low $v_k$, 0.1%, and a high intensity, 99%. It has three cluster centres, and the highest data reduction is 27%. These three cluster centres have a suitable geometry, so the cost data can be partitioned into relevant contents to be redacted for imputing. Our model shows better clustering detection and evaluation than models using k-means. The percentage of missing data for the two cost objects is the following: labour 57%, materials 81%. The second level shows highly correlated data ($R$-
squared 0.99) after imputing. Therefore, the study concludes multi-objective GA can cluster and impute data to derive complete data for forecasting.

**Subjects:** Optimization; Linear & Nonlinear Optimization; Artificial Intelligence

**Keywords:** Data clustering; data imputing; multi-objective GA; fuzzy $c$-means; K-means clustering

1. Introduction

Data clustering is used to divide data as a set of objects into groups based on their similarities (Handl & Knowles, 2007; Runkler, 2012) and their sharing of relevant properties (Xu & Wunsch, 2005). Clustering is useful to explore the interrelationships between data points (Jain, Murty, & Flynn, 1999). An evaluation function can represent the data mathematically as sets, partition matrices, and/or cluster prototypes (Runkler, 2012). An evaluation function is able to measure the intrinsic properties of a partition of the data based on the compactness of the clusters and the separation between them (Handl & Knowles, 2007) to determine the quality of the partitioning (Handl & Knowles, 2007).

A difficulty of clustering is defining appropriate data clusters using multiple cluster validity indices to achieve multi solutions rather than a single solution (Deb, 2014; Saha & Bandyopadhyay, 2010). The goal of such clustering is to structure data for use in scale analysis (Xu & Wunsch, 2008). Longitudinal studies posit the existence of missing values to decrease the computational process, estimate model variables, and derive the results that would have been seen if the complete data were used. The common practice is to impute the missing data using the average of the observed values. With imputing, no values are sacrificed, thus precluding the loss of analytic results (Schafer & Graham, 2002).

A multi-objective genetic algorithm (GA) is a widely used evaluation technique to optimise clustering (Deb, Pratap, Agarwal, & Meyarivan, 2002). Multi-objective GA optimises particular functions to determine automatically the number of clusters and the partitions in a dataset simultaneously and to compute their fitness (Saha & Bandyopadhyay, 2010). It can predict missing data by finding an approximate solution interval that minimises the error prediction function (Lobato et al., 2015). Several studies of data clustering and imputing have used multi-objective GAs to understand and improve data to avoid bias in decision-making. For example, Siripen Wikaisuksakul proposed a multi-objective GA for automatic clustering in an $n$-dimensional database. The database is assigned a number of fuzzy groups in different dimensions using the fuzzy $c$-means (FCM) algorithm. This method can handle overlapping in automatic clusters by exploiting the fuzzy properties’ advantages. However, the model needs more computation time to process unlabelled data without prior knowledge (Wikaisuksakul, 2014).

Julia Handl et al. proposed a new data clustering approach, in which two or more measures of cluster quality are simultaneously optimised using a multi-objective evolutionary algorithm. The authors adapted Pareto envelope method selection for the clustering problem by incorporating specialised mutation and initialisation optimisation procedures for a number of objectives. They found this clustering approach yields high quality solutions for data clustering and greater understanding of data structure. The approach may incorporate feature selection, outlier-removal, and fixed cluster numbers (Handl & Knowles, 2004).

Lecai Cai et al. proposed a novel $k$-means clustering algorithm based on immune GA to improve the genetic operators for searching and clustering. They found the proposed algorithm can improve searching by adapting to different data types. Because of the high number of GA repetitions, however, the algorithm is complex (Cai, Yao, He, & Liang, 2010). Fabio Lobato et al. used multi-objective GA (MOGAImp) to impute missing data. Their model is suitable for tackling conflicting evaluation measures; it can be used for mixed-attribute datasets and takes the information
from incomplete data into account by modelling. The authors suggest investigating the MOGAImp application for clustering or advanced analysis to reduce the search space and extract the knowledge required to build a model (Lobato et al., 2015). Their technique does not decrease the search space complexity.

The aim of this study is to develop a new two-level multi-objective GA to optimise clustering and impute the missing cost data of tunnel fans. The first level clusters cost data into the most relevant components and redacts the data. The second level imputes the missing cost data to create complete data for forecasting and analysis. The first level uses a multi-objective GA based on a FCM algorithm to optimise cluster centres. The selected cluster centres are optimised by considering the following: (1) cluster outliers, (2) compactness, variation between data points and clusters, and separation between clusters, and (3) intensity of data points belonging to selected clusters. The clustered data can be redacted to the relevant data points without redundancy or noise. Our module is validated using k-means clustering. In the second level, a multi-objective GA imputes the missing redacted cost data from the first level to derive complete data for forecasting. We argue that a multi-objective GA decreases the complexity, increases the flexibility, and is very effective when selecting an approximate solution interval for imputing.

2. Data collection
The study uses cost data for tunnel fans installed in Stockholm in Sweden. The data were collected over 10 years, from 2005 to 2015, by Trafikverket and are stored in a MAXIMO computerised maintenance management system (CMMS). In this CMMS, the cost data are recorded based on the work orders for the maintenance of the tunnel fans. Every work order contains corrective maintenance data, a component description, the reporting date, a problem description, and a description of the actions performed. Also included are the repair time used and the labour, material and tool cost of each work order.

In this study, we consider two cost objects, i.e. labour and materials, based on the work order input into the CMMS for the 10-year period mentioned above. We did not use the tool cost data because of the huge numbers of missing data. The selected data were clustered, filtered and imputed for the study using a multi-objective GA based on a FCM algorithm. It is important to mention that all the cost data are for real costs with no adjustment for inflation. Due to company regulations, all the cost data are encoded and expressed as currency units (cu).

3. K-means clustering
K-means algorithm divides N points in M dimensions into K clusters so that data points within the same cluster have a minimum sum of squares. The general procedure is to search for a K-partition with locally optimal within cluster by moving points from one cluster to another. The number of points in cluster C is denoted by Euclidean distance between the data point and cluster itself (Hartigan & Wong, 1979; Lohrmann & Luukka, 2018).

The data set consists of \( N = (1, 2, \ldots, N) \) data points. Three clusters \( \nu_i = (1, 2, \ldots, C) \) selected from the work orders of cost data portion the data into M disjoint subsets (cluster). Three clusters assumed based on many literatures. The sum of the squared Euclidean distance is used to calculate the distance between data point \( N \) and the cluster \( \nu_i \) as seen in the formula below (Likas, Vlassis, & Verbeek, 2003):

\[
E(\nu_i) = \sum_{j=1}^{N} \sum_{k=1}^{C} N_j - \nu_k^2
\]

4. Two-level system of multi-objective genetic algorithms (GAs)
The study develops a new two-level multi-objective GA, as shown in Figure 1. The GA consists of: (1) a multi-objective GA based on FCM to cluster cost data and (2) a multi-objective GA to impute missing data. In level 1, the GA is applied to the cost data objects (labour, material) at six different
times (six populations) to cluster and redact data for the next level. The second level longitudinally imputes the missing data of the three cost objects. The use of two levels allows us to reduce noisy data and computational cost (Thomassey & Happiette, 2007), whilst reaching an effective and reasonable solution (Ding, Cheng, & He, 2007).

4.1. Level 1: multi-objective GA based on fuzzy c-means algorithm
A multi-objective GA is widely applied in clustering because of its ability to optimise cluster centres in a large space of random populations (Deb et al., 2002). The GA operates with a population of chromosomes containing data of work orders. The cluster centres are selected randomly from the population of chromosomes. The chromosomes are proportional to the case and problem statement (Cordón, Herrera, Gomide, Hoffmann, & Magdalena, 2001).

A multi-objective GA operates on a population of cluster centre chromosomes over different generations. Over generations, chromosomes in the population are rated on their adaptation (Sortrakul, Nachtmann, & Cassady, 2005) to the new population and on their selection mechanism. Their adaptability, evaluated as the fitness function, creates the basis for a new population of chromosomes. The new population is formed using specific genetic operators, such as crossover and mutation. The fitness function is derived from the membership function (Cordón et al., 2001; Piroozfard, Wong, & Wong, 2018) using the FCM algorithm. The membership function gives cluster centre chromosomes a possible solution for the generational problem of adapting (Wikaisuksakul, 2014).

The FCM algorithm is a broad spectrum algorithm which tries to subdivide a dataset N into C pairwise disjoint subsets (cluster centres) (Bezdek, Ehrlich, & Full, 1984). This algorithm can assign data to multiple clusters. The degree of membership in the fuzzy cluster centres depends on the
closeness of a given data point to a cluster centre (Izakian & Abraham, 2011). The FCM estimates the fuzzy partitions of the data and the locations of cluster centres (Hathaway & Bezdek, 2001; Lei et al., 2018). Random selection in FCM cluster centres is an iterative process yielding optimal local solutions. A multi-objective GA can process the clustering (Jain et al., 1999) by optimising the search for local optimum data points.

This paper describes a multi-objective GA based on FCM implemented six different times using a cross-validation randomisation technique. The technique aims to find the optimal cluster centre geometry with respect to the shaping of data points around cluster centres. The process is the following: a random number of cluster centres is generated randomly in each of the six implementations; the modified cluster centres are generated 25 times. The modifications are used to find the geometry of cluster centres in the cost data and determine the optimal ones. Twenty-five generations are enough for this study to obtain valid results. The following steps explain the use of the multi-objective GA.

**Step 1: initial population**

The cost data are a sequence of maintenance work orders; each work order is described as a chromosome and consists of a description of the work order with two cost objects: labour and materials ($Z_{labor}$, $Z_{material}$). The description is useful to understand the relevant data points belonging to the cluster centres.

**Step 2: first multi-objective GA generation**

In the first generation, the weights of each cost object are calculated in sets of $N = \{1, 2, \ldots, N\}$ data points using the Modified Adaptive Weight Approach (MAWA). MAWA is selected because it can simplify data points and efficiently calculate multi-objective GA. To calculate the weights of each cost object, we find the minimum and maximum value for each, $Z_{labor} = \{Z_{labor, min}, Z_{labor, max}\}$, $Z_{material} = \{Z_{material, min}, Z_{material, max}\}$, and $Z_{tool} = \{Z_{tool, min}, Z_{tool, max}\}$. The minimum and maximum values determine the relevant MAWA formula. Then, each data point in the cost object is calculated based on the conditions, as seen in the formulas below. The formulas are modified from Zheng, Ng, and Kumaraswamy (2004) to accommodate different conditions:

$$\text{MAWA}(z) = \frac{Z_{min} (Z_{max} - z_c) + \gamma (Z_{max} - Z_{min})}{(Z_{max} - Z_{min}) + \gamma}$$

when $Z_{max} \neq Z_{min}$

$$\text{MAWA}(z) = 0.1 \times \frac{(Z_{max} - z_c) + \gamma (Z_{max} - Z_{min})}{(Z_{max} - Z_{min}) + \gamma}, \text{when } Z_{min} = 0$$

$$\text{MAWA}(z) = 0.9 \times \frac{(Z_{max} - z_c) + \gamma (Z_{max} - Z_{min})}{(Z_{max} - Z_{min}) + \gamma}, \text{when } Z_{max} = Z_{min}$$

$\gamma$: positive random number between 0 and 1; $z_c$: cost of the $z$th values in the current population of the object.

MAWA represents three different conditions to guide the multi-objective GA and widen the search scope. The first condition, wherein the maximum equals the minimum of the object, the weight is multiplied by 0.9. The second condition, wherein the minimum equals zero, the weight is multiplied by 0.1. The third condition, wherein the minimum does not equal the maximum, the weight depends on the current values. In these latter cases, weights can be calculated with different data types based on the maximum and minimum value for each individual object. The overall weights, as seen in the equation below, are the summation of the MAWAs of each object. This summation decreases the time required to calculate the membership function, as the evaluation has one object instead of three.

$$\text{MAWA}_{overall} = \sum_{m=1}^{x-1} \text{MAWA}(z)$$
Step 3: selection
The selection of cluster centres (chromosomes) uses the chromosome pool and is directed by the concept of natural genetic systems. A random number of cluster centres, \(v_i\), is generated between 2 and \(\sqrt{N}\). This enables the proposed clustering technique to evolve the appropriate partitioning data automatically with clusters of any shape, size, or convexity, as long as they possess some point symmetry property (Saha & Bandyopadhyay, 2010). \(N\) is the number of data points in the cost object. Cluster centres \(v_i = \{v_1, v_2, v_3, \ldots, C\}\) are selected and removed from \(N\) data. The selected chromosomes contain MAWA for each cost object and MAWAoverall. Noise is a random number between 0 and 1 and is added to MAWAoverall to avoid zero distance and find the relations between data points and cluster centres. The calculated fitness function defines the membership of a data point in the cluster centre. To calculate this, we need to calculate the square of the Euclidean distance \(d_{ik}^2\) between data points \(z_i\) and \(v_i\) (Bezdek et al., 1984).

\[
d_k = \sum_{j=1}^{j-1} (z_j - v_j)^2, \quad 1 \leq k \leq N, \quad 1 \leq i \leq C
\]

\(z_j\): object’s data points; \(v_j\): cluster’s centre points.

Fitness function membership, fitness \((z_k)\), is the degree of belonging of \(z_i\) to \(v_i\) (Wikaisuksakul, 2014) and is expressed as:

\[
\text{fitness}(z_k) = \frac{1}{\sum_{j=1}^{C} \left(\frac{d_k}{d_j}\right)^{2/(m-1)}}, \quad 1 \leq k \leq N, \quad 1 \leq i \leq C
\]

The highest fitness \((z_k)\) of a data point will be in cluster centre \(v_i\).

The first objective of the paper is to calculate cluster centre outliers to select the best pool of cluster centres. Given the equal fitness function of each data point to the selected cluster centres, \(v_1, v_2, v_3, \ldots, C\), the number of outliers should be small. In this case, the cluster outlier problem, or the error of the membership degree \(J(z,v)\) function, is expressed as the equal distance between data point \(z\) and cluster centres \(v_i\) (Bezdek, 1973).

\[
J(z,v) = \sum_{i=1}^{C} \sum_{k=1}^{N} \text{fitness}(z_k) d_{ik}^2
\]

\(C\): number of cluster centres; \(N\): number of data points belonging to cluster centres.

The second objective of the paper is to select the best cluster centres based on compactness and separation percentages, \(v_i (z, v, N)\). Compactness measures the variation in data within a cluster or between data and cluster centroids; this variation should be kept small. Separation considers the variation between cluster centroids; this variation should be large. To obtain the best cluster centres over GA generations and to find the optimal number of clusters, we need to find the minimum percentage of \(v_i\) (S.H. Kwon, 1998).

\[
v_k(z, v; N) = \frac{\sum_{i=1}^{C} \sum_{k=1}^{N} \text{fitness}(z_k) d_{ik}^2 + \left(\frac{1}{2} \sum_{i=1}^{C} \sum_{k=1}^{N} \|v_i - v_j\|^2 + \gamma \right)}{\min_{ij} \|v_i - v_j\|^2 + \gamma}
\]

\(\|v_i - v_j\|^2\): average distance between \(C\) cluster centres; \(\min_{ij} \|v_i - v_j\|^2\): minimum distance between \(C\) cluster centres.

The average distance is used to eliminate the monotonically decreasing tendency when there are large numbers of clusters; \(\gamma\) is a random value between 0 and 1 to avoid division by zero.
The third objective is to measure the intensity, that is, the number of data points \( z_i \) belonging to cluster centres \( v_i \). This will show the cluster centres’ geometry.

Overall, these objectives are intended to facilitate the measurement of the aggregation of fuzzy membership degrees so that we may evaluate and optimise the cluster centres and find the best cluster centres in the population.

Step 4: crossover and mutation
Crossover is a probabilistic process exchanging information between two parent cluster centres (chromosomes) to generate two new child cluster centres (chromosomes). In this study, we use single point crossover with a fixed crossover probability to generate new cluster centre values because we only use two data objects (labour, material). For cluster chromosomes of length \( l \), a crossover point is generated in the range \([1, l-2]\). The chromosomes to the right of the crossover point are exchanged to produce two cluster centre offspring. The objects are connected by weight, and the value of the weight should move with the new cluster generation.

Fifty percent of the collected cluster centres randomly experience mutation. As we are considering the representation of data point values in this study, we swap between costs values for each selected work order of chromosomes after crossover.

Step 5: new generation
The new generation of cluster centres will repeat steps 3–4 continuously for 25 generations. At this point, the best fitting GA generation of cluster centres will be selected to cluster and redact data in volume with keeping the same data parameter’s structure. Twenty-five generations are enough for these data because the fitness function value is repeated after the fifth generation. The new redacted data will not contain noise or redundant contents. They will mostly have relevant and significant data points belonging to cluster centres and should differ according to cost data description. The resulting data will be ready for the second level, imputing the missing data.

4.2. Level 2: multi-objective genetic algorithm
In this level, a multi-objective GA is applied longitudinally to the data. Multi-objective GA operates with a population of chromosomes that contains labour cost and materials cost objects. Forty percent of each cost object is selected randomly at two different times. Multi-objective GA operates on the selected population over different generations to find the appropriate cost range to impute missing data for each object. During the GA generations, chromosomes in the population are rated on their adaptation, and their selection mechanism to the new population is evaluated. Their adaptability (fitness function) is the basis for a new population of chromosomes. The new population is formed using specific genetic operators such as crossover and mutation. Multi-objective GA is used to fill in the zero value data from an appropriate cost range based on the best fitness function over different generations. The following steps describe level 2.

Step 1: initial population
A longitudinal study of each cost object (\( Z_{labour}, Z_{material}, Z_{tool} \)) is used to impute missing data using multi-objective GA for the three objects in parallel.

Step 2: first GA generation and selection
The first generation is done by randomly selecting 40% of the initial population of cost objects at two different times so that 80% of the data comprise different chromosomes. We select two chromosomes to validate the GA operators. We select 80% because a high range of missing data requires us to find an appropriate range of values to impute missing data.

We find the minimum and maximum for 40% of the population and for each cost object. We generate a random number \( P_i \) between the minimum and maximum of each cost object. Data
point $z_i$, fitness ($z_i$) for each selection in each object is calculated based on the formula below (Hadavandi, Shavandi, & Ghanbari, 2011). Then, the chromosome pairs for each object are summed.

The fitness function is a statistical method finding the root mean absolute percentage error of the correlated imputed data. The lowest fitness function is selected to transform data to the next generation after the GA is applied.

$$
\text{fitness}(z_i) = \sqrt{\frac{1}{N} \sum_{t=1}^{N} (Y_i - P_i)^2}
$$

$N$: number of training data; $Y_i$: actual value of $i$th training; $P_i$: generated value of $i$th training.

**Step 3: crossover and mutation**

In this study, we use a two-point crossover with a fixed crossover probability. For chromosomes of length $l$, a crossover point is generated in the range $[1, 1/3 \ l]$, $[1/3 \ l, 2/3 \ l]$, and $[2/3 \ l, \ l]$. The values of objects are connected and should be exchanged to produce two new offspring. We select two points to create more value ranges and find the best fit.

Fifty percent of the selected chromosomes randomly undergo mutation with the arrival of new chromosomes. For the cost object values, we swap two opposite data values.

**Step 4: new generation**

The new generation will repeat steps 2–3 continuously for 25 generations. Twenty-five generations are enough for these data because the fitness function increases after the fourth generation. The best fit generation will have the lowest fitness function. The selected generation is used to randomly impute new values between the selected ranges for the missing data for each cost object. This step will yield fully correlated data for better estimation of the forecasting model.

After this step, $R$-squared regression analysis is used to validate the imputed cost objects. $R$-squared gives information about the correlations of the cost values for each cost object before and after imputing.

5. Results and discussion

In what follows, we present the results of our two-level multi-objective GA.

### 5.1. Results of level 1: multi-objective GA based on fuzzy c-means algorithm

In this study, we test six populations individually using a multi-objective GA based on a FCM algorithm and compare them to optimise the cluster centres for the three different cost objects. Figure 2 shows the number of random cluster centres generated for each population with the actual cluster centres around which data points are shaped. The actual cluster centres of the generated clusters are the clusters around which the data points are shaped. In addition, the actual cluster centres (chromosomes) are modified over 25 generations of the data population. The first and second population have three actual cluster centres, created from 17 and 12 random cluster centres respectively. The third population has only one cluster centre, derived from eight random cluster centres. The fourth, fifth and sixth populations have between two and three cluster centres from 16, 17, and 14 random cluster centres respectively. The number of cluster centres is not the main criterion; rather, we are more concerned with the optimal cluster centres. The percentage of intensity for each population gives a better understanding of the cluster centres' effectiveness.

As Figure 3 shows, in the first population, 98.80% of the data belong to three cluster centres. The second population has two cluster centres, and 99.15% of the data are relevant. The third population has the highest percentage with 99.43% relevant data, but it has only one cluster
centre. In this case, we cannot prove the population is optimal because one cluster centre is not
enough to understand the data. The fourth and fifth populations have 98.88% and 98.8% relevant
data respectively. The fifth population has only two cluster centres (the fourth has three), so it has
the lowest percentage. The sixth population has 99% of the data shaped around three cluster
centres. The optimal cluster centre will have the most data points shaped around it.

Cluster centre roles can be represented by the percentage of data reduction for each population.
Figure 4 shows the data reduction percentages for each population after choosing and implement-
ing the cluster centres.

As Figure 4 shows, the first and second populations have 26.55% and 26.33% data reduction in
volume respectively. The first population has three cluster centres and the second has two. The
first has the highest reduction percentage over the six populations but the lowest intensity. The
third population has 26.11% reduction, the lowest percentage, but there is only one cluster centre
in this population, not enough to understand the data. The fourth and fifth populations have 26.4%
reduction for three and two cluster centres respectively. The last population has 26.25% reduction
with three cluster centres and 99% intensity. Overall, the data reduction percentages are not high enough to determine the optimal cluster centres. This means we need to study compactness and separation ($\nu_k$) for the cluster centres of each population.

Figure 5 shows the $\nu_k$ percentages for each population. In this figure, a lower $\nu_k$ means lower compactness and higher separation between cluster centres. The first population has a $\nu_k$ of 0.09% over 25 generations, the lowest value. This population has three cluster centres and higher data reduction, as seen in Figures 2 and 4. The second population has higher $\nu_k$ than the first population, at 0.25%, and two cluster centres. The third has 0.11% $\nu_k$ with one cluster centre. In this case, one cluster is excluded because the separation in $\nu_k$ is zero, so it cannot be compared with other populations. The fourth and fifth populations have 1.17% and 0.09% $\nu_k$, respectively, with two and three cluster centres, respectively, as seen in Figure 2. The highest $\nu_k$ is found in the fourth population; here, the cluster centres are almost convergent. The sixth population has the highest $\nu_k$, 3.49%, with three cluster centres among the six populations.
The optimal population with optimal cluster centres does not always mean the population and the cluster centres have the lowest and highest peaks for the number of cluster centres, the intensity, the data reduction percentage, or $\nu_k$. In this study, we average the optimal cluster centres to discover most of the data points; specifically, we look at efficient intensity, high data reduction, and low $\nu_k$ of cluster centres. The error membership $J(z, \nu)$ has zero value in the proposed methodology over populations and with different multi-objective GA generations.

In this study, the first population has a low $\nu_k$, 0.09%, a high percentage of intensity, 98.80%, with three cluster centres, and the highest data reduction, 26.55%, compared to other populations. These three cluster centres have a suitable geometry in the data, making most of the cost data points understandable and discoverable. Figure 6 shows the data points for the two cost objects, labour and materials, and the clustering results of the first generation with three cluster centres.

These cluster centres are used to reduce the data into relevant parts and prepare for the second level of multi-objective GA to impute the missing data. Note that Figure 6 has three different colours for the three cluster centres of the first population. The clusters distributed over the data points because the cluster centres geometry stands on optimal place. These centres make better understanding of the data points. The data are compared and evaluated to reduce the original data to the most relevant contents without redundancy or noise.

5.2. Results of level 2: genetic algorithm
The outcome of the first level, specifically in the first population, indicates the amount of missing data in the cost objects. Labour and materials costs have 56.84% and 81.02% missing data respectively. Missing data cause a substantial amount of bias, make the analysis of the data more arduous, and reduce analysis efficiency. Multi-objective GA is implemented for each cost object to impute the missing data. The imputation will help to provide complete data for better estimation of forecasting models.

$R^2$-squared regression analysis is used to validate the imputation process using multi-objective GA and to determine the differences before and after imputation for each cost object. Figure 7 shows $R^2$-squared is 0.847 for the reduced accumulated labour cost data, and Figure 8 shows a better $R^2$-squared value, 0.9926, after imputing the missing labour cost data. This means imputing increases the correlation of the labour cost values. Figure 9 shows $R^2$-squared is 0.4401 for the accumulative materials cost before imputation; after the missing data are imputed, Figure 10 shows a higher $R^2$-squared result, 0.9996. The correlation between cost values is improved after imputation.
Using multi-objective GA for each level is time consuming, even when it executes on a virtual machine of a server with special specifications. The server used in this study has 16 processors of E5-2690V2 (25M cash), 128 GB of RAM, 300 GB of Hard disk, and Windows-7 Enterprise 64 bit. Table 1 shows the time taken for clustering and imputation for each population. Clearly, clustering is more time consuming than imputation.

Unlike imputation, the time taken for clustering is dissimilar for each population; the clustering process has different calculations and comparisons based on a randomly generated number of cluster centres. The imputation process has fewer calculations and requires almost the same amount of time for each population, as the population size is always 80% of the cost data for each cost object.
6. Validation
A k-means algorithm is implemented on cost data objects to cluster data. Figure 11 shows the data points for the two cost objects, labour and materials, and the clustering results for three clusters. These clusters divide the data points into three certain partitions. The certain cluster numbers might be inaccurate, and the data need more or less cluster. In addition, the clustering does not show an understanding of the relations and correlations between the data points for labour and material costs. The data reduction will not be easy, because a cluster’s geometry might not be the optimal one.

7. Conclusions
This study develops a new two-level multi-objective GA based on a FCM clustering algorithm. This GA creates a large search space for optimising cluster centres to reduce data. The algorithm gives us the ability to evaluate cost data and find suitable clustering for a problem statement. In addition, it performs
well with poor data with high missing values. The optimal cluster centres are the average cluster centres with efficient intensity, high data reduction, and low $\nu_k$. Multi-objective GA based on FCM has these advantages. In addition, our clustering model shows better results than $k$-means clustering.

The two-level multi-objective process is time consuming, however, and requires a special server and specifications. In addition, the algorithm needs further research to derive a rule-based system for clustering and to reduce the time required. Multi-objective GA imputing shows an optimal range of values to impute missing data. It can provide appropriate data ranges to accommodate data variations, and it shows minimum imputing errors. The correlation of cost data values after the missing data are imputed is high and relevant.

Acknowledgements
The authors gratefully acknowledge Dr. Ali Ismail Awad, Luleå University of Technology, Luleå, Sweden, for his support and for allowing us to use the computing facilities of the Information Security Laboratory to conduct the experiments in this study. This work is partially supported by the Youth Innovative Talent Project (2017KQNCX191) from the Department of Education of Guandong Province, China.

Funding
The authors received no direct funding for this research.

Author details
Yamur K. Aldouri
E-mail: yamur.aldouri@ltu.se
ORCID ID: https://orcid.org/0000-0002-1967-6604
Hassan Al-Chalabi
E-mail: hussan.hamodi@ltu.se
Liangwei Zhang
E-mail: liangwei.zhang@dgut.edu.cn
1 Division of Operation and Maintenance Engineering, Luleå University of Technology, Luleå, Sweden.
2 Department of Industrial Engineering, School of Mechanical Engineering, Dongguan University of Technology, Dongguan, China.

Citation information
Cite this article as: Data clustering and imputing using a two-level multi-objective genetic algorithm (GA): A case study of maintenance cost data for tunnel fans, Yamur K. Aldouri, Hassan Al-Chalabi & Liangwei Zhang, Cogent Engineering (2018), 5: 1513304.

References

Table 1. Time taken for each population

<table>
<thead>
<tr>
<th>GA populations</th>
<th>Clustering (HH:MM:SS)</th>
<th>Imputation (HH:MM:SS)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10:00:51</td>
<td>00:00:16</td>
</tr>
<tr>
<td>2</td>
<td>05:38:48</td>
<td>00:00:17</td>
</tr>
<tr>
<td>3</td>
<td>02:16:58</td>
<td>00:00:17</td>
</tr>
<tr>
<td>4</td>
<td>08:56:05</td>
<td>00:00:16</td>
</tr>
<tr>
<td>5</td>
<td>09:15:17</td>
<td>00:00:17</td>
</tr>
<tr>
<td>6</td>
<td>06:18:05</td>
<td>00:00:17</td>
</tr>
</tbody>
</table>


Handl, J., & Knowles, J. (2004). Evolutionary multobjec-


Wikalsusakul, S. (2016). A multi-objective genetic algo-

rithms. IEEE Transactions on Neural Networks, 16(3), 645–678. doi:10.1109/TNN.2005.845141

