INCOMPLETE ANALYTIC HIERARCHY PROCESS (AHP) SOLUTION METHODOLOGIES

by

DOĞUKAN ZORLU

Submitted to the Graduate School of Engineering and Natural Sciences

in partial fulfillment of

the requirements for the degree of Master of Science

SABANCI UNIVERSITY

July 2023

ABSTRACT

INCOMPLETE ANALYTIC HIERARCHY PROCESS (AHP) SOLUTION METHODOLOGIES

DOĞUKAN ZORLU

Industrial Engineering, MSc. Thesis, July 2023

Thesis Supervisor: Assoc. Prof. Dr. Kemal Kılıç

Keywords: Analytic Hierarchy Process, Incomplete Information, Decision Making, Pairwise Comparisons, Completion Methodologies

Analytic hierarchy process (AHP) is a well-known multi criteria decision making method. It relies on matrices constructed through pairwise comparisons of criteria and alternatives by decision makers. In the literature, AHP has been proven effective in assessing weights of criteria and/or relative scores of alternatives. However, in order to utilize the AHP method, it is necessary for the decision maker to determine complete matrices, meaning that the decision maker must assign all pairwise comparisons into the matrices. This requirement may not always be feasible due to reasons such as the decision maker's inadequate knowledge in certain pairwise comparisons, uncertainty in the pairwise comparisons, and time constraints. In the literature, several algorithms have been proposed to address this issue, but there is no consensus on the best algorithm. In this study, we provided a comparative analysis of the existing algorithms. Furthermore, we also introduced new parametric heuristic algorithms for the incomplete AHP framework. The proposed algorithms were also compared with the existing algorithms in different experimental conditions. The performances of these methods were assessed utilizing metrics from the literature as well as a metric developed for the incomplete AHP framework in this research. The comparisons were conducted in two types of experiments, i.e., numerical and empirical. According to the results of these experiments, the developed algorithms were competitive, and even demonstrated better performance under some experimental conditions.

ÖZET

TAMAMLANMAMIŞ ANALİTİK HİYERARŞİ PROSESİ (AHP) ÇÖZÜM METODOLOJİLERİ

DOĞUKAN ZORLU

Endüstri Mühendisliği, Yüksek Lisans Tezi, Temmuz 2023

Tez Danışmanı: Assoc. Prof. Dr. Kemal Kılıç

Anahtar Kelimeler: Analitik Hiyerarşi Süreci, Eksik Bilgi, Karar Verme, İkili Karşılaştırmalar, Tamamlama Yöntemleri

Analitik hiyerarşi prosesi (AHP) çok kriterli karar verme yöntemi olarak bilinen bir yöntemdir. AHP, karar vericiler tarafından kriterler ve alternatifler arasında ikili karşılaştırmalarla oluşturulan matrislere dayanır. Literatürde, bu yöntemin kriterlerin ağırlıklarını ve/veya alternatiflerin göreceli puanlarını değerlendirmede etkili olduğu kanıtlanmıştır. Fakat, AHP yönteminden yararlanmak için karar vericinin tam matrisler oluşturması gereklidir, yani karar vericinin bütün ikili karşılaştırmaları matrislere ataması gerekir. Bu gereklilik, uzmanın bazı ikili karşılaştırmalarda yetersiz bilgiye sahip olması, ikili karşılaştırmalardaki belirsizlik ve zaman sınırlaması gibi nedenlerden dolayı her zaman mümkün olmayabilir. Literatürde, bu sorunu ele almak için birkaç algoritma önerilmiştir, fakat en iyi algoritma konusunda fikir birliği yoktur. Bu çalışmada, mevcut algoritmaların karşılaştırmalı bir analizini sunduk. Ayrıca, tamamlanmamış AHP çerçevesi için yeni parametrik sezgisel algoritmalar da tanıttık. Tanıtılan algoritmalar, farklı deneysel koşullarda mevcut algoritmalarla da karşılaştırılmıştır. Bu yöntemlerin performansları, literatürdeki metriklerin yanı sıra bu araştırmada tamamlanmamış AHP çerçevesinde geliştirilen bir metrik kullanılarak değerlendirilmiştir. Karşılaştırmalar, sayısal ve deneysel olmak üzere iki tür deneyde gerçekleştirilmiştir. Bu deneylerin sonuçlarına göre, tanıtılan algoritmalar rekabetçi bir performans sergilemiş ve bazı deneysel koşullar altında daha iyi performans göstermiştir.

Dedicated to my beloved family,

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATONS

AHP	Analytic Hierarchy Process
CI	Consistency Index
CR	Consistency Ratio
PCM	Pairwise Comparison Matrix
MCIV	Modified Compatibility Index Value
GCIV-VW	Generic Compatibility Index between V and W
SLLS	Sparse Logarithmic Least Squares
TLT	Transitivity of Length of Two
TLET	Transitivity of Length of not Exceeding Three
NEC	Number of Entries that Change
DEMATEL	Decision-Making and Trial Evaluation Laboratory
Tukey HSD	Tukey's Honest Significant Difference

1. INTRODUCTION

Decision making deals with the process of determining and selecting alternatives according to the preferences of decision makers. It should consider the pros and cons of each choice. It is a crucial phenomenon since people are always making decisions in their social lives, business, and economics. However, determining the priority of each alternative can be challenging, especially when the number of alternatives is enormous, and subjectivity is an issue. This challenge arises from the ambiguity of the alternatives or complexity of the decision-making process itself. In order to address this challenge, decision makers might rely on knowledge of the relative importance of alternatives rather than on extracting their exact value. Some researchers have suggested that eliciting judgments on two alternatives separately is easier than eliciting judgments on all alternatives simultaneously. Choo et al. (2016) proposed that the use of pairwise comparison is the preferred methodology to extract human preferences, as this approach evaluates options in a binary manner, making it less cognitively demanding than evaluating all options at once. This consideration might be all the more important considering the work of Miller (1956), who suggested that the human brain stores a limited amount of information which includes between five and nine items in working memory.

Another common issue is transitiveness. In the ideal scenario data, which comes from decision makers are transitive, meaning that if alternative A has a higher utility than that of B and B has a higher utility than that of C, then A has a higher utility than that of C. Note that consistency and transitivity are interconnected concepts. The presence of consistency leads to transitivity. Unfortunately, real-world situations are complicated by perturbations, noise, or subjective biases, which make it challenging to achieve consistency and transitivity (Davis 1958, Saaty 1977, Bessi et al. 2015).

One of the effective methodologies that deals with the decision-making phenomenon even if the above challenges are present is analytic hierarchy process (AHP), which was introduced by Saaty (1972, 1977). AHP is a decision-making methodology designed to support decision makers when dealing with complex problems that includes multiple subjective and conflicting criteria (Ishizaka and Labib, 2011). According to Emrouznejad and Marra (2017), AHP is considered to be a leading decision-making approach since it includes subjective factors to be taken into account. AHP has been successfully applied in a wide range of fields such as, warehouse network evaluation (Korpela and Lehmusvaara 1999), supplier selection (Chamodrakas et al., 2010), project selection (Amiri, 2010), the health sector (Saaty and Vargas, 1998), marketing (Wind and Saaty, 1980), university evaluation (Lee, 2010), human resource manager selection (Kusumawardani and Agintiara, 2015), and other various domains.

AHP is a very flexible approach and can handle many situations, some of which have already been mentioned. However, it has several limitations, some of which are ranking reversal, complexity, sufficient expertise knowledge, and incomplete data. The ranking reversal issue refers to a situation in which the relative ranking of alternatives changes when new alternatives are added or deleted. In other words, the ranking of alternatives is not consistent. The complexity issue may arise from the nature of the decision-making problem itself. When the problem includes numerous criteria and alternatives, the pairwise comparisons might become more complex and time-consuming. The correctness and reliability of AHP is highly related to the decision maker who is the expert of the subject under consideration. A lack of expertise in a decision maker may lead to inconsistent or unreliable results. Missing data may cause a major issue in AHP since consistency and accuracy highly depend on complete and precise data on the pairwise comparisons. Several solutions to these challenges have been published in AHP literature. However, they are out of the scope of this research except for the issue of incomplete data in AHP.

Handling incomplete data in pairwise comparison matrices has been a frequently discussed topic in the AHP literature. Several studies have suggested a methodology which copes with the incompleteness issue (e.g., Harker, 1987a; Harker, 1987b; Carmone et al., 1997; Bozóki et al., 2010; Gomez-Ruiz et al., 2010; Oliva et al., 2017; Zhou et al., 2018; Menci et al., 2018; Oliva et al., 2018). These studies can be classified into two subsets: reconstructing methods and sparse methods. Reconstructing methods aim to fill

in the missing entries in the pairwise comparison matrix, and then identify the rankings and priorities. In contrast, sparse methods utilize the pairwise comparison matrix as it is, without modification, to determine the rankings and priorities. Although there are many studies on reconstructing methods in the AHP literature (Harker, 1987a; Carmone et al., 1997; Bozóki et al., 2010; Gomez-Ruiz et al., 2010; Zhou et al., 2018), sparse methods have also been the subject of recent research (Oliva et al., 2017; Menci et al., 2018; Oliva et al., 2018).

There are several reasons why pairwise comparison matrix in AHP can be incomplete, such as time complexity, insufficient knowledge or vagueness about expertise in certain comparisons, and loss of collected data. In AHP, it is sufficient to conduct n(n-1)/2, where "n" represents the alternative number, pairwise comparisons per criterion since AHP assumes that the data collected from decision makers is reciprocal. This means that if A is rated as three times better than B, then B must be rated as one-third as good as A. Moreover, the diagonal entries of the pairwise comparison matrix must be one since each element is equivalent to itself. Thus, it is sufficient to complete one triangular section of the matrix (ex. upper right triangular). When the alternatives or criteria number are huge, it may be inefficient or time consuming to gather all of the necessary entries. When the decision maker has not established a strong perspective on certain assessments, it may be preferable to let them skip the question rather than compel them to make an unreliable estimate. Carmone et al. (1997) conducted an experimental study in which entries are removed from matrices having various sizes of 10, 15, 20. The results of the study demonstrated that it is possible to eliminate up to 50% of the entries in the pairwise comparison matrix without causing a significant decrease in the outcome.

When determining priorities, the goal is to establish a consistent matrix whose entries represent the ratios of one priority to another. The entries in this matrix should match the pairwise comparison matrix collected from the decision maker. Moreover, it should have minimal deviation when slight inconsistencies are present. Several approaches to derive these priorities can be found in the AHP literature. According to Saaty (1977), the preferred priorities can be determined by the principal eigenvector $\boldsymbol{w}$. Following Saaty's innovative work, several methods for deriving priority vectors have been proposed. One of these was created by Crawford and Williams (1985). These authors addressed the issue of rank reversal in the eigenvector method and adapted an alternative technique called the logarithmic least squares method.

The incomplete AHP literature has also benefited from work in other disciplines to address the issues of calculating missing entry or determining priorities. Harker (1987b) suggested an approach for filling the missing entries based on the concept of connecting path. This approach involves calculating all indirect comparisons based on the transitivity rule and taking the geometric mean of these comparisons to assess the missing entry. Zhou et al. (2018) utilized decision-making and trial evaluation laboratory (DEMATEL) methodology and adapted it to reconstruct an incomplete pairwise comparison matrix. Olivia and colleagues (2017) adopted the eigenvector introduced by Saaty to sparse context to derive priorities. In 2018, Menci and colleagues introduced three alternative methodologies for finding priorities in sparse settings. These techniques are based on well-known methods in the several literatures including Metropolis-Hastings Markov chains (Metropolis et al., 1953), Heat-Bath Markov chains (Achlioptas et al., 2005), and formation control (Fax and Murray, 2004).

Evaluating the accuracy of priorities is another important issue. There are various evaluation metrics available in the literature, including consistency index, consistency ratio, and compatibility index. However, when using these indices, which were designed for complete pairwise comparison matrices, on incomplete pairwise comparison matrices, it is essential to consider the assumptions and meanings behind them. For instance, in a sparse setting where some entries are zero, the compatibility index becomes meaningless because the elementwise product of the missing entries yields 0 which reduces the compatibility index. The decline in the compatibility index is not due to the integrity of the assigned values, but rather it arises from the emptiness of the matrix. Furthermore, using the consistency index of an incomplete matrix is inappropriate since it assumes a complete matrix. Therefore, when the pairwise comparison matrix is incomplete, it may be necessary to develop new metrics or modify existing ones to evaluate priorities accurately.

This research focuses on incomplete pairwise comparison matrix methodologies in the context of AHP. Some of the key accomplishments of this study are as follows:

1. Novel parametric heuristic algorithms were created to handle an incomplete pairwise comparison matrix in the context of AHP.

2. A new metric, that is suitable for incomplete context, was developed. The relationships between the newly introduced metric and the metrics that are suitable in incomplete AHP framework were analyzed, and correlations were investigated.

3. Methodologies for handling incomplete pairwise comparison matrices in incomplete AHP literature were statistically compared among themselves and with the proposed algorithms by several metrics under varying experimental designs.

4. Methodologies were evaluated through both numerical and empirical studies.

The rest of this thesis is organized as follows. Chapter 2 covers preliminary definitions and notations. Chapter 3 provides a review of existing algorithms in the literature. Chapter 4 introduces proposed parametric heuristic algorithms and explains the motivation behind them. The numerical and empirical experimental designs are also introduced in chapter 4. Chapter 5 presents, results and discussion. Chapter 6 concludes this research and discusses future research areas.

2. GENERAL NOTATIONS AND PRELIMINARIES

2.1. General Notations

Number of alternatives are demonstrated as *n*. Vectors are represented using italic, boldface, and lowercase letters (e.g., vector v is shown as $\boldsymbol{v}$), whereas matrices are denoted using italic and uppercase letters (e.g., matrix A is shown as *A*). Additionally, the $(i,j)^{th}$ entry of a matrix *A* is represented by A_{ij}, and the i^{th} entry of a $\boldsymbol{v}$ is represented by $\boldsymbol{v_i}$. If we have a vector $\boldsymbol{v}$ that belongs to $\mathbb{R}^n$, then D(v) is an $n \times n$ diagonal matrix where the i^{th} diagonal entry is $\boldsymbol{v_i}$. An identity matrix is denoted by I_n. Furthermore, A ○ B represents the Hadamard product, which is the element-wise product of two matrices *A* and *B* that have the same dimensions.

2.2. Preliminaries

Definition 1: Analytic hierarchy process (AHP) is a multi-decision-making method that was developed by Saaty (1972, 1977). In general, AHP involves four main steps including problem modelling, pairwise comparison matrix creation, priority weights calculation and consistency measurement. In the first step a hierarchical structure is created by breaking down the decision problem into smaller components. The second step involves creating pairwise comparison matrices (PCMs) to assess the relative importance of alternatives. Each element in the PCM is usually determined by using a Saaty scale of values ranging from 1 to 9 (Table 1). In the third step the priority weights of the alternatives are calculated based on the PCMs. In the last step, the consistency of the PCMs is measured for

reliability. The main objective of AHP is to prioritize the alternatives and allocate weight to each of them in accordance with the pairwise comparisons.

Table 1 Saaty scale

Definition	Intensity of Importance
Equal Importance	1
Moderate Importance	3
Strong Importance	5
Demonstrated Importance	7
Extreme Importance	9
Intermediate Values	2, 4, 6, 8

Definition 2: Pairwise Comparison Matrix (PCM) Let X be a $n \times n$ PCM, where X_{ij} represents the relative importance of i^{th} criteria on j^{th} criteria. Moreover, X should satisfy the following conditions.

$$X_{ij} > 0 \text{ and } X_{ij} = \begin{cases} 1/X_{ji} & if\ i \neq j \\ 1 & if\ i = j \end{cases} \tag{2.1}$$

In the context of incomplete AHP, it can be modified as follows.

$$A_{ij} = \begin{cases} X_{ij} & if\ it\ is\ known \\ 1 & if\ i = j \\ 0 & otherwise \end{cases} \tag{2.2}$$

Definition 3: True Priority Vector $\boldsymbol{v} = [\boldsymbol{v_1}, \boldsymbol{v_2}, \ldots, \boldsymbol{v_n}]$ is the ideal depiction of the decision maker's preferences. It can be used as a benchmark for evaluating the performance of the process.

Definition 4: Calculated Priority Vector $\boldsymbol{w} = [\boldsymbol{w_1}, \boldsymbol{w_2}, \ldots, \boldsymbol{w_n}]$ is the weight of the alternatives derived by a methodology. According to Saaty (1977) $\boldsymbol{w}$ is reproduced by the principal right eigenvector which corresponds to the maximum eigenvalue of the PCM (Equation 2.3). The main objective is to extract a weight vector that closely mates the true priority vector. In other words, to minimize the deviation between the calculated priority vector and the true priority vector. Large deviation can be seen as inconsistency in the process.

$$Xw = \lambda_{max}\, w \tag{2.3}$$

Definition 5: Consistency Index (CI) and Consistency Ratio (CR) were characterized by Saaty.

The Consistency Index is defined in equation 2.4.

$$\mathrm{CI} = \frac{\lambda_{max} - n}{n - 1} \tag{2.4}$$

The Consistency Ratio is expressed below.

$$\mathrm{CR} = \frac{\mathrm{CI}}{\mathrm{RI}} \tag{2.5}$$

where RI is the Random Index, which is the average of the consistency indexes obtained from randomly generated PCMs. It depends on the dimension of PCMs and some of the values of RI are shown below (Table 2) (Hayrapetyan, 2019). If CR $\leq$ 0.1 then the derived PCM can be acceptable, which means PCM is sufficiently consistent. If CR = 0, PCM is fully consistent (Saaty, 1977).

Table 2 Random index values

n	1	2	3	4	5	6	7	8	9	10
RI	0	0	0.52	0.89	1.13	1.25	1.35	1.43	1.47	1.5

Definition 6: Theoretical Pairwise Comparison Matrix (*W*) is a $n \times n$ matrix which is calculated from ***w.*** The value of W_{ij} is the ratio of i^{th} entry of ***w*** over j^{th} entry of ***w.***

Definition 7: True Theoretical Pairwise Comparison Matrix (*V*) is a $n \times n$ matrix which is constructed from ***v.*** V_{ij} is the ratio of i^{th} of ***v*** entry over j^{th} entry of ***v***.

Definition 8: Kendall's Correlation Index (Kendall's tau, τ) is an index, which was created by Kendall (1938), for identifying the degree of relevance between two vectors (***v, w***) based on ordinal or ranked data. It evaluates the strength of the association between these vectors. Kendall's tau can change between -1 and 1. If $\boldsymbol{\tau} = +1$, it shows that ***v*** and ***w*** have the identical order. On the other hand, if $\boldsymbol{\tau} = -1$, it means that they have opposite order. In other words, if tau is closer to 1, it shows that two vectors are correlated. If tau is closer to -1, it represents that two vectors are anti-correlated. Lastly, if tau is closer to 0, it means that two vectors have no correlation.

Two pairs of values ($\boldsymbol{v_i}$, $\boldsymbol{w_i}$) and ($\boldsymbol{v_j}$, $\boldsymbol{w_j}$) are concordant if both $\boldsymbol{v_i} < \boldsymbol{v_j}$ and $\boldsymbol{w_i} < \boldsymbol{w_j}$ or both $\boldsymbol{v_i} > \boldsymbol{v_j}$ and $\boldsymbol{w_i} > \boldsymbol{w_j}$. Two pairs of values ($\boldsymbol{v_i}$, $\boldsymbol{w_i}$) and ($\boldsymbol{v_j}$, $\boldsymbol{w_j}$) are discordant if both $\boldsymbol{v_i} < \boldsymbol{v_j}$ and $\boldsymbol{w_i} > \boldsymbol{w_j}$ or both $\boldsymbol{v_i} > \boldsymbol{v_j}$ and $\boldsymbol{w_i} < \boldsymbol{w_j}$. Moreover, if $\boldsymbol{v_i} = \boldsymbol{v_j}$ or $\boldsymbol{w_i} = \boldsymbol{w_j}$ the pairs are considered to be neither concordant nor discordant. Let $\boldsymbol{v}$ and $\boldsymbol{w}$ belongs to $\mathbb{R}^n$, then $\boldsymbol{\tau}$ is calculated as follow.

$$\boldsymbol{\tau} = \frac{2(|C| - |D|)}{n(n-1)} \tag{2.6}$$

where C represents the set of concordant pairs and D represents the set of discordant pairs.

Definition 10: Compatibility Index Value (CIV), which is a metric described by Saaty (1994), demonstrates the deviation between *X* and *W*. CIV is defined in equation 2.7.

$$\text{CIV} = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n} X_{ij}\frac{\boldsymbol{w_j}}{\boldsymbol{w_i}} \tag{2.7}$$

Note that, when a matrix *X* is fully consistent, *X* and *W* are exactly equal. This means that the CIV of matrix *X* and *W* would be equal to 1. On the other hand, if matrix *X* is inconsistent, CIV will be greater than 1 (Saaty, 1994)

Definition 11: Modified Compatibility Index Value (MCIV), is modified version of CIV, is created for incomplete AHP setting in this research. It measures only the deviation between the cells assigned by the decision maker and the corresponding entries of *W*. Let *m* be the number of the missing data of PCM. Then, MCIV is described as below.

$$\text{MCIV} = \frac{1}{n^2-m}\sum_{i=1}^{n}\sum_{j=1}^{n} A_{ij}\frac{\boldsymbol{w_j}}{\boldsymbol{w_i}} \tag{2.8}$$

Definition 12: Generic Compatibility Index between *V* and *W* (GCIV-VW), was defined by Ahmed and Kilic (2022). It illustrates the deviation between *V* and *W*. As in CIV, GCIV-VW also equals 1 if fully consistency presents. In practical application matrix *V* is unknown, so that matrix *X* and/or *W* were usually used to assess the calculated priority vectors in the literature. However, in an empirical and a numerical application true priority vector may be known. In such a context, GCIV-VW metric can be utilized to assess methodologies. GCIV-VW is demonstrated as follows.

$$\text{GCIV} - \text{VW} = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}\frac{\boldsymbol{w_i}}{\boldsymbol{w_j}}\frac{\boldsymbol{v_j}}{\boldsymbol{v_i}} \tag{2.9}$$

Definition 12: Euclidean Distance (*Euc*), measures the distance between two objects in a space. Euclidean distance is commonly used in many areas such as statistics, data analysis and machine learning. Euclidean distance is characterized as follows.

$$Euc = \sqrt{(\boldsymbol{w_1} - \boldsymbol{v_1})^2 + (\boldsymbol{w_2} - \boldsymbol{v_2})^2 + \cdots + (\boldsymbol{w_n} - \boldsymbol{v_n})^2} \quad (2.10)$$

Definition 13: Central Processing Unit Time, is a metric which evaluates the time for which a central processing unit (CPU) was employed for processing commands of operating system.

2.2.1. Graph Theory

Let G be a graph, represented as G = (V, E) where V is a set of nodes (vertices), and E is a set of edges. Node i are shown as $\boldsymbol{v_i}$ and edge between node i and node j are indicated as ($\boldsymbol{v_i}$, $\boldsymbol{v_j}$). An edge from $\boldsymbol{v_i}$ to $\boldsymbol{v_j}$ in E indicates that there is a connection between them. A graph is undirected if for any edge ($\boldsymbol{v_i}$, $\boldsymbol{v_j}$) in E, ($\boldsymbol{v_j}$, $\boldsymbol{v_i}$) must present in E. In a connected graph, there must be at least one path between any pair of nodes in G such that all nodes on that path must be connected by edges. In a connected graph there are no disconnected subsets of nodes. In the context of AHP, it is a common practice to assume PCM is undirected because of the reciprocal property of PCM. In the continuation of this research, the connectedness of PCM is another assumption since some methodologies in incomplete AHP literature encloses it such as the Metropolis Hastings and the Heat Bath algorithms developed by Menci et al. (2018). Therefore, numerical and empirical experiments were analyzed accordingly.

An adjacency matrix of a graph with *n* nodes is a *n* × *n* matrix, represented by *M*. Adjacency matrix is a fundamental concept since it provides analysis of the connectivity of a graph and degree of nodes. The entries of *M* are either 1 or 0.

$$M_{ij} = \begin{cases} 1 & \text{if } there\ is\ an\ edge\ between\ \boldsymbol{v_i}\ and\ \boldsymbol{v_j} \\ 0 & otherwise \end{cases} \quad (2.11)$$

The neighborhood of node i, denoted as N(i), is identified as the set of nodes that are adjacent to node i. For an undirected graph it can be represented as follows.

$$N(i) = \{\boldsymbol{v_i}, \boldsymbol{v_j} \in V: (\boldsymbol{v_i}, \boldsymbol{v_j}) \in E\} \quad (2.12)$$

For an undirected graph, the degree of node i, represented as *deg(i)*, is the number of nodes which are adjacent to node i. It can be shown as equation 2.13.

$$deg(i) = |\{\boldsymbol{v_i}, \boldsymbol{v_j} \in V: (\boldsymbol{v_i}, \boldsymbol{v_j}) \in E\}| \quad (2.13)$$

Moreover, it can be found by utilizing adjacency matrix of an undirected graph as follows.

$$deg(i) = \sum_{\boldsymbol{v_j} \in V} M_{ij} \quad (2.14)$$

The degree matrix of a graph, denoted as *DG*, is a diagonal matrix whose entries equal the degree of the nodes or 0, represented as below.

$$DG_{ij} = \begin{cases} deg(i) & \text{if i} = \text{j} \\ 0 & otherwise \end{cases} \quad (2.15)$$

The degree vector of a graph, denoted as $\boldsymbol{d}$, whose entries equal the degree of the nodes. A $n \times n$ matrix L is irreducible if there is no P, which is a $n \times n$ permutation matrix whose rows and columns having exactly one 1 and other entries are 0, such that,

$$P^T L P = \begin{bmatrix} L_{11} & L_{12} \\ 0 & L_{22} \end{bmatrix} \quad (2.16)$$

where L_{11} is $m \times m$, L_{22} is $(n - m) \times (n - m)$, L_{12} is $n \times (n - m)$, and m is an integer with $0 < m < n$. Moreover, in the case of undirected graphs, a matrix L is irreducible if and only if its structure corresponds to a connected graph G (Oliva et al., 2018).

2.2.2. Markov Chains

The definitions presented in the Markov chains will be helpful in comprehending certain methodologies utilized to solve the sparse setting, especially in chapters 3.4 and 3.5.

Definition 14: Discrete Time Markov Chain (DTMC), let Z be a finite or countable set of the Markov chain, which is called state space of the chain. A DTMC is a sequence of random variables $X_0, X_1, \ldots$ taking values in the set Z, satisfying the following equation.

$$P(X_{n+1} = x_{n+1} \mid X_0 = x_0, \ldots X_{n-1} = x_{n-1}, X_n = x_n) = P(X_{n+1} = x_{n+1} \mid X_n = x_n) \quad (2.17)$$

for all $x_0, \ldots, x_n, x_{n+1} \in Z$ and $n \in N$, where P(X | Y) defines the conditional probability of X given Y. In other words, a DTMC is a stochastic model that defines a series of events, where the likelihood of each future state depends solely on the current state and not on any of the previous states.

Definition 15: Single Step Transition Matrix (*P*), let Z contains *n* states. Transition matrix is a $n \times n$ square matrix which consists of the probabilities of moving between states of the chain in a single time unit. Let P_{ij} defines the probability of moving from state i to state j in a single step. Furthermore, the transition matrix must include two properties (2.18 and 2.19).

$$P_{ij} \geq 0 \tag{2.18}$$

$$\sum_{j=1}^{n} P_{ij} = 1 \tag{2.19}$$

Definition 16: Time Homogeneous Markov Chain, a Markov chain is time-homogeneous if the transition probabilities constant over time, i.e., it satisfies the following condition,

$$P(X_{n+1} = j \mid X_n = i) = P(X_1 = j \mid X_0 = i) \text{ for all } n \in N \tag{2.20}$$

In time homogeneous MC,

$$p(X_{n+1}) = P^T p(X_n) \tag{2.21}$$

where $p(X_n)$ represents probability distribution of X_n.

For homogenous DTMC, let the n-step transition probability of being in state j given that the chain was in state i, is represented as P_{ij}^n and n-step transition matrix is denoted as P^n. According to Chapman-Kolmogorov equations (Dobrow, 2016),

$$P^{(n+m)} = P^n P^m \quad \text{for m, n} \geq 0 \tag{2.22}$$

Definition 17: Irreducible Markov Chain, is a Markov chain where all the states in the state space communicate with each other (single class). State i is accessible from state j if P_{ij}^n is positive for some $n \geq 0$. Any state i and j in the state space communicate each other if both are accessible from each other. Moreover, irreducibility of transition matrix, yields irreducibility of Markov chain.

Definition 18: Period of Markov Chain, the period of state i, denoted as d(i), is defined as follows.

$$d(i) = \gcd\{n > 0: P_{ii}^n > 0\} \tag{2.23}$$

where "gcd" refers to the greatest common divisor. If d(i) = 1, then state i is said to be aperiodic. Note that $P_{ii} > 0$ is an adequate condition for aperiodicity. When a Markov

chain is irreducible, and all states are aperiodic then Markov chain is called aperiodic Markov chain. Moreover, periodicity is a class property, i.e., if state i communicates with state j and is periodic/aperiodic, then state j is also periodic/aperiodic. Therefore, when a Markov chain is irreducible, there is a single class in which all states have the same periodicity property. Hence, if single step transition matrix is irreducible and for any i in state space has $P_{ii} > 0$, then Markov chain is aperiodic.

Definition 19: Limiting Distribution (π) of a Markov chain is a probability distribution such that,

$$\lim_{n\to\infty} P_{ij}^{n} = \pi_j \tag{2.24}$$

In other words, the probabilities of being in each state converge after some steps to a certain value which is independent of the initial distribution of the states.

Note that, if a Markov chain with finite state space is aperiodic and irreducible, then it has a limiting distribution.

Remark 1: Suppose an irreducible but not aperiodic Markov chain with n states, let P is its transition matrix. Then for any $\beta \in (0,1)$, matrix $\beta P + (1-\beta)I_n$ is aperiodic and its left dominant eigenvector is the same with P (Menci et al., 2018).

3. METHODOLOGIES IN INCOMPLETE AHP LITERATURE

In experimental designs several incomplete AHP methodologies in literature namely Harker (1987a), Bozóki et al. (2010), Oliva et al. (2017), Zhou et al. (2018) and Menci et al. (2018) were compared. These incomplete AHP methodologies in literature are summarized in this chapter.

3.1. Harker

Harker (1987a) created a new matrix *B* and used its principal right eigenvector to assign missing entries of *A*.

The derivation can be depicted as follows.

1. Establish a matrix *B* as,

$$B_{ij} = \begin{cases} 0 & if\ A_{ij}\ is\ missing \\ 1 + \mathrm{m}_i & if\ i = j \\ A_{ij} & otherwise \end{cases} \tag{3.1}$$

   where m_i represents the number of missing entries in the i'th row.

2. Calculate the principal right eigenvector $\boldsymbol{w}$ and eigenvalue $\boldsymbol{\lambda_{max}}$ of *B*.
3. Fill the missing entries of *A* by using $\boldsymbol{w}$ (if A_{ij} is missing, assign it as $\boldsymbol{w_i/w_j}$)

3.2. DEMATEL

Zhou et al. (2018) proposed a DEMATEL based solution methodology to derive priorities in the context of incomplete AHP. DEMATEL is a kind of structural modelling technique, applicable to examine the cause-and-effect relationships between alternatives. It can be applied effectively to extract the interrelationships among alternatives within a complex system. Moreover, it can provide the ranking of these alternatives (Si et al., 2018). It has demonstrated its effectiveness in various fields, including but not limited to risk assessment (Li et al., 2020), supply chain management (Wu et al., 2017), and stock selection (Shen and Tzeng, 2015). There are two important matrices in DEMATEL including DRM and TRM. DRM is direct relation matrix, and it contains direct relations. TRM is total relation matrix, and it includes both indirect and direct relations. DEMATEL consists of five steps namely determining quality characteristics, deriving the DRM, normalizing DRM, constructing TRM and classifying factors. Zhou et al. (2018) adapted DEMATEL methodology into AHP framework in four steps. The purpose is extracting complete pairwise comparison matrix using incomplete pairwise comparison matrix. In the first step, the incomplete pairwise comparison matrix is converted into DRM and then DRM is normalized. This conversion is generated as follows. Let *A* be the incomplete PCM, *D* be the DRM.

$$D_{ij} = \begin{cases} A_{ij} & if\ A_{ij}\ \ is\ assigned\ by\ decision\ maker \\ 0 & otherwise \end{cases} \tag{3.2}$$

After the creation of DRM, the sum of each row and column is calculated for the normalization step. The maximum of the row sum and column sum is identified and is utilized to normalize DRM, which is denoted as *ND.* Normalization step is shown as in equation 3.3.

$$ND = \frac{D}{\max\left(\sum_{i=1}^{n} D_{ij}\ ,\ \sum_{j=1}^{n} D_{ij}\right)} \tag{3.3}$$

In the second step TRM, denoted as *T*, is created by using *ND* as below.

$$T = \lim_{m\to\infty}(ND + ND^2 + \cdots + ND^m)$$

$$= \lim_{m\to\infty}(ND\ (I_n\ - ND^m)(I_n\ - ND)^{-1})$$

$$= ND(I_n\ - ND)^{-1} \tag{3.4}$$

Zhou et al. (2018) emphasized that the progression from 1 to infinity represents the gradual discovery of indirect relationships between each pair in DRM. Sometimes TRM may not be calculated as above formula because of the nonexistence of inverted matrix (singularity of matrix). For these situations Zhou et al. (2018) designed an experiment to see the convergence of $\lim_{m\to\infty}(ND + ND^2 + \cdots + ND^m)$. According to their experiment, they observed that convergence of limit is very quick (m is around 5 in most cases). They suggested that it is appropriate to set m to be a specific integer like m = 5 and approach TRM with that assumption. In the experimental designs in this study, we approached TRM as utilizing coefficient of 25.

In the third step of their methodology, a complete pairwise comparison matrix is constructed from TRM with satisfying reciprocal property. The construction is made as following algorithm.

Algorithm 1: Conversion from TRM to PCM

1: **Input:** the matrix $T_{n\times n}$

2: **Construct:** a matrix $P_{n\times n}$

3: **for** i =1, …, n **do**

4: **for** j =1, …, n **do**

5: $P_{ij} = \sqrt{\frac{T_{ij}}{T_{ji}}}$

6: $P_{ji} = \frac{1}{P_{ij}}$

7: **Output:** *P*

DEMATEL based solution methodology abbreviated as Dematel in the continuation of this research.

In the last step the missing values of A are derived by using P as follows.

$$A_{ij} = \begin{cases} A_{ij} & if\ A_{ij}\ was\ assigned\ by\ the\ decision\ maker \\ P_{ij} & otherwise \end{cases} \quad (3.5)$$

Note 1: Following 4 methodologies are related with the sparse setting. In the context of sparse setting PCM is filled differently. In the nominal case, the diagonal entry of the PCM is assigned as 1, but in the sparse context it is assigned as 0. The missing entries also filled as 0 in the sparse setting. Therefore, while constructing the degree matrix (DG), the diagonal entries of PCM are not considered.

Example 1: Let PCM is assigned as follows.

$A = \begin{bmatrix} 1 & - & 2 & 0.5 \\ - & 1 & 3 & - \\ 0.5 & 0.34 & 1 & 0.25 \\ 2 & - & 4 & 1 \end{bmatrix}$, where "-" represents the missing value.

Then, in the sparse context, it is demonstrated as below.

$A = \begin{bmatrix} 0 & 0 & 2 & 0.5 \\ 0 & 0 & 3 & 0 \\ 0.5 & 0.34 & 0 & 0.25 \\ 2 & 0 & 4 & 0 \end{bmatrix}$

Furthermore, its degree matrix is considered as below.

$DG = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix}$

Note 2: The connectedness of the G, which is derived from A, is an assumption for the following 4 methodologies in order to calculate the priorities.

3.3. Sparse Eigenvector

The motivation behind the Sparse Eigenvector method developed by Oliva et al. (2017) is coming from the following result. When there is missingness but no perturbation, entries in the PCM are transitive and accurate, the principal right eigenvector of $DG^{-}A$ is the same as the true priority vector $\boldsymbol{v}$. Based on this outcome, they suggested using the principal right eigenvector of $DG^{-}A$ as calculated priority vector $\boldsymbol{w}$. The ideology resembles the eigenvector method created by Saaty.

3.4. Metropolis-Hastings

Menci et al. (2018) constructed the Metropolis-Hastings method to extract priorities in the case of the sparse setting. Their motivation behind the methodology is inspired by the Metropolis Hastings Markov chain. They suggested constructing a matrix *P*, which has the same structure as the connected graph G, utilizing *A*. The derivation is as follows.

$$P_{ij} = \begin{cases} \dfrac{min\{1, \frac{\boldsymbol{d_i} A_{ji}}{\boldsymbol{d_j}}\}}{\boldsymbol{d_i}} & if\ (\boldsymbol{v_i},\ \boldsymbol{v_j})\ \in \text{E} \\ 1 - \frac{1}{\boldsymbol{d_i}} \sum_{\text{k}\in \text{N(i)}} min\{1, \frac{\boldsymbol{d_i} A_{ki}}{\boldsymbol{d_k}}\} & if\ i = j \\ 0 & otherwise \end{cases} \tag{3.6}$$

where $\boldsymbol{d}$ is the degree vector and N(i) is the neighborhood of node i.

Similar to the Sparse Eigenvector methodology, when there is no perturbation the left dominant eigenvector of *P* gives the true priority vector $\boldsymbol{v}$ (Menci et al., 2018).

Note that the matrix *P* is a form of transition probability matrix of a Markov chain by construction. It holds the following two properties (2.18 and 2.19).

Because G is undirected, connected and contains finite states, the Markov chain is irreducible since there is a single class, and all states can communicate with each other. When there is no perturbation, the Markov chain is aperiodic, so that the Markov Chain

has limiting distribution, which is determined by the left dominant eigenvector (Menci et al., 2018). The limiting distribution was taken as the calculated priority vector $\boldsymbol{w}$.

However, when dealing with perturbations, it cannot be guaranteed that the Markov chain will exhibit aperiodicity (Menci et al., 2018). In order to ensure convergence, a modified version of the Markov chain was employed by them. Therefore, they used convex combination of P^T and I_n as shown below.

$$\bar{P} = \beta P^T + (1-\beta) I_n, \text{ where } \beta \in (0,1) \tag{3.7}$$

Note that irreducibility property is not affected by convex combination. Moreover, $\bar{P}$ is aperiodic and its left dominant eigenvector is the same with P (Remark 1). Therefore, the new Markov chain has a limiting distribution, which is used for approaching the true priority vector $\boldsymbol{v}$. The algorithm they proposed for reaching the limiting distribution of P is as follows.

Note 3: $\boldsymbol{w_i}$(k +1) represents the probability of the Markov chain will be in state i state at the step k+1.

Algorithm 2: Metropolis-Hastings

1: **Input:** the matrix $A_{n \times n}$, random parameter β

2: **Obtain:** the matrix $P_{n \times n}$, and the vector $\boldsymbol{d}$

3: **Initialization**

4: **for** i =1, …, n **do**

5: $\boldsymbol{w_i}(0)$ = random (0, 1)

6: **Standardize** $\boldsymbol{w}(0)$

7: **Synchronous Iteration**

8: $\boldsymbol{w_i}(k+1) = (\beta P_{ii} + 1 - \beta)\, \boldsymbol{w_i}(k) + \beta \sum_{j \in N(i)} P_{ji}\, \boldsymbol{w_j}(k)$

9: **Output:** $\boldsymbol{w}$

Note that, in their research they proposed more than one method for initialization procedure. However, in this research standardized random positive rational numbers were utilized. Initialization is necessary for determining the initial condition of the Markov chain. The aim of the synchronous iteration is approaching the limiting distribution. After several iterations the change between iterations becomes tiny. In this research, 0.000001

was used as a terminating condition. If all states change became less than 0.000001, the algorithm stopped, and $\boldsymbol{w}$(k+1) was taken as calculated priority vector. Moreover, in experimental setups parameter β was generated as in equation 3.12.

$$\beta = \text{random}(0, 1) \tag{3.8}$$

3.5. Heat-Bath

Menci et al. (2018) created the Heat-Bath method in order to derive priorities for the sparse setting. Their motivation behind the methodology gets inspired by Heat-Bath Markov chain.

They proposed forming a matrix *P* that is adapted from *A* as below. It can be shown that *P* has the same structure as the graph G.

$$P_{ij} = \begin{cases} \frac{\gamma}{1 + A_{ij}} & if\ (\mathrm{i}, \mathrm{j})\ \in\ \mathrm{E} \\ 1\ -\ \gamma \sum_{\mathrm{k} \in\ \mathrm{N(i)}} \frac{1}{1 + A_{ik}} & if\ i = j \\ 0 & otherwise \end{cases} \tag{3.9}$$

where γ is a random parameter which must satisfy the following condition.

$$\gamma < \frac{1}{max\{\boldsymbol{d_i}\}} \tag{3.10}$$

Note that similar to the Metropolis-Hasting method the matrix *P* is a form of transition probability matrix by construction. It is known that G is undirected, connected and contains finite states by assumption. Therefore, the Markov chain is irreducible. Moreover, since the diagonal entries are positive, it is aperiodic.

$$\sum_{\mathrm{j} \in \mathrm{N(i)}} P_{ij} = \gamma \sum_{\mathrm{j} \in \mathrm{N(i)}} \frac{1}{1 + A_{ij}} < \frac{1}{max\{\boldsymbol{d_i}\}} \sum_{\mathrm{j} \in \mathrm{N(i)}} 1 \leq 1 \tag{3.11}$$

Consequently, since the row sum is 1 all diagonal entry must satisfy the below condition.

$$P_{ii} > 1\ -\ \frac{\boldsymbol{d_i}}{max\{\boldsymbol{d_i}\}} \geq 0 \tag{3.12}$$

Therefore, there exists a limiting distribution $\boldsymbol{w}$. When there is no perturbation, this limiting distribution becomes the true priority vector (Menci et al., 2018). On the other hand, in the existence of perturbations they suggested approaching the true priority vector

by the limiting distribution $\boldsymbol{w}$. The algorithm they proposed for attaining the limiting distribution of P is as below.

Algorithm 3: Heat-Bath

1: **Input:** the matrix $A_{n \times n}$, random parameter γ

2: **Construct:** the matrix $P_{n \times n}$

3: **Initialization**

4: **for** i =1, …, n **do**

5: $\boldsymbol{w_i}(0)$ = random (0, 1)

6: **Standardize** $\boldsymbol{w}(0)$

7: **Synchronous Iteration**

8: $\boldsymbol{w_i}(k+1) = \sum_{j \in N(i) \cup \{i\}} P_{ji}\ \boldsymbol{w_j}(k)$

9: **Output:** $\boldsymbol{w}$

Note that, they recommended several methodologies for initialization procedure. In this research, standardized random positive rational numbers were used. The goal of synchronous iteration is to reach the limiting distribution. As the iterations progress, the changes gradually diminish. In the experimental designs, a value of 0.000001 was employed as a termination criterion. When the change in all states became less than 0.000001, the algorithm terminated, and $\boldsymbol{w}(k+1)$ became the calculated priority vector. Furthermore, in experimental designs parameter γ was generated as below.

$$\gamma = \text{randomuniform}(0, \frac{1}{max\{\boldsymbol{d_i}\}}) \tag{3.13}$$

3.6. Sparse Logarithmic Least Squares

The latest approach within the sparse setting is the sparse logarithmic least squares (SLLS), which was introduced by Menci et al. (2018). This algorithm is inspired by a widespread method logarithmic least square (LLS) developed by Crawford (1987). As an adaptation, the SLLS algorithm aims to approach the true priority vector $\boldsymbol{v}$. In other words, the objective is detecting a vector ($\boldsymbol{w}$) which ensures the log quadratic minimization of

the error between pairwise comparison matrix *A* and theoretical pairwise comparison matrix *W*. The following equation (3.14) demonstrates the derivation of $\boldsymbol{w}$.

$$\boldsymbol{w} = arg\ min_{x \in \mathbb{R}_+^n} \{\sum_{\mathrm{i}}^{n} \sum_{\mathrm{i} \in \mathrm{N(i)}}^{n} (\ln A_{ij} - \ln(\frac{\boldsymbol{x_i}}{\boldsymbol{x_j}}))^2\} \tag{3.14}$$

The algorithm they used to approach such $\boldsymbol{w}$ is shown below. For further information, e.g., the theorem and the derivation behind the algorithm, see Menci et al. (2018).

Algorithm 4: Sparse Logarithmic Least Squares

1: **Input:** β

2: **Initialization**

3: **for** i =1, …, n **do**

4: $\boldsymbol{r_i}(0)$ = random positive real number

5: $\boldsymbol{w_i}\ (0)$ = expo ($\boldsymbol{r_i}\ (0)$)

6: $\boldsymbol{s_i} = \sum_{\mathbf{j} \in \mathbf{N(i)}}^{\boldsymbol{n}} \ln A_{ij}$

7: **Synchronous Iteration**

8: $\boldsymbol{r_i}(\mathrm{k+1}) = \boldsymbol{r_i}(\mathrm{k}) + \beta \sum_{\mathbf{j} \in \mathbf{N(i)}} (\boldsymbol{r_j}(\mathbf{k}) - \boldsymbol{r_i}(\mathbf{k})) + \beta \boldsymbol{s_i}$

9: $\boldsymbol{w_i}(\mathrm{k+1}) = \mathrm{expo}(\boldsymbol{r_i}(\mathrm{k+1}))$

10: **Output:** $\boldsymbol{w}$

Note that, in the experimental designs in this research a value of 0.000001 and 30000 (70000 was used only in one experiment since in that experiment convergence was slow) were used as a termination criterion. When the change in all states became less than 0.000001 or the changes could not be less than 0.000001 in 30000 iterations, the algorithm stopped and returned $\boldsymbol{w}$(k+1) as the calculated priority vector. Furthermore, Menci et al. (2018) suggested to employ β such that,

$$\beta \leq \frac{1}{max\{\boldsymbol{d_i}\}} \tag{3.15}$$

Therefore, in this research β was chosen as below.

$$\beta = \text{randomuniform } (0, \frac{1}{max\{\boldsymbol{d_i}\}}) \tag{3.16}$$

Note that the logarithmic least squares method is widely used in AHP literature both in standard setting and incomplete setting. For example, in the incomplete case Menci et al.

(2018) and Bozoki et al. (2010) utilized logarithmic least square as objective. Therefore, both methodologies can reach the same calculated priority vector. The difference between them is the algorithm they used. For instance, Menci et al. was inspired by the Fax and Murrays formation control algorithm (Fax and Murray, 2004; Olfati-Saber et al., 2007)

4. PARAMETRIC HEURISTIC ALGORITHMS, THE NUMERICAL AND EMPIRICAL STUDY EXPERIMENTAL DESIGN

4.1. Parametric Heuristic Algorithms

A natural approach to complete the missing element in the pairwise comparisons matrix is to take the geometric average of all the indirectly calculated comparisons of missing entry (Harker, 1987b). However, a limitation of this approach is that as the number of alternatives increases, the number of indirect comparisons grows significantly, which results in a time complex. The parametric heuristics proposed in this article derive its motivation from this idea. It decreases the time complexity by using some of the indirect comparisons. Moreover, some of the proposed heuristics weigh these indirect comparisons based on their lengths. By length, it is meant to how many arcs the path has. In order to achieve indirect comparisons, it utilizes graph theory. Paths which reach missing values by transitivity are examined. After all the paths leading to the missing value are found, the missing value is calculated by taking the geometric average of the value of these paths. The value of the paths is calculated as the product of its arcs. One advantage of using the geometric average is that it preserves the reciprocal property of the matrix, which is an assumption of AHP.

Let A_{ij} and A_{ji} be the missing value and the paths from i to j having values as z_1, z_2, …, z_k. It is known that the assigned values from decision maker preserve reciprocal property and path length is calculated as product of its arcs, thus any path from i to j has the inverse value of the path from j to i if they use the same arc set but in reverse order. Moreover, if there is a path having specific arc set, the existence of the reverse order is guaranteed

since the graph is undirected. Then the paths from j to i having values as $1/z_1$, $1/z_2$, …, $1/z_k$. Therefore, the geometric average of the missing values is reciprocal.

Example 2: Finding missing value by paths and geometric average methodology.

Let the pairwise comparison matrix be as follows, where "-" represents the missing value.

$$A = \begin{bmatrix} 1 & - & 5 & 0.5 & 2 \\ - & 1 & - & 0.25 & 0.5 \\ 0.2 & - & 1 & 0.25 & - \\ 2 & 4 & 4 & 1 & 5 \\ 0.5 & 2 & - & 0.2 & 1 \end{bmatrix}$$

Paths having length less than or equal to three that start from node 1 and end with node 2, length and value are as below.

P1: $A_{13} - A_{34} - A_{42}$, length: 3, value: $5 \times 0.25 \times 4 = 5$

P2: $A_{14} - A_{42}$, length: 2, value: $0.5 \times 4 = 2$

P3: $A_{14} - A_{45} - A_{52}$, length: 3, value: $0.5 \times 5 \times 2 = 5$

P4: $A_{15} - A_{52}$, length: 2, value: $2 \times 2 = 4$

P5: $A_{15} - A_{54} - A_{42}$, length: 3, value: $2 \times 0.2 \times 4 = 1.6$

Therefore, A_{12} is: $\sqrt[5]{5 \times 2 \times 5 \times 4 \times 1.6} = 3.17$

Paths having length less than or equal to three that start from node 2 and end with node 1, their length and value are as below.

P6: $A_{24} - A_{43} - A_{31}$, length: 3, value: $0.25 \times 4 \times 0.2 = 0.2$

P7: $A_{24} - A_{41}$, length: 2, value: $0.25 \times 2 = 0.5$

P8: $A_{25} - A_{54} - A_{41}$, length: 3, value: $0.5 \times 0.2 \times 2 = 0.2$

P9: $A_{25} - A_{51}$, length: 2, value: $0.5 \times 0.5 = 0.25$

P10: $A_{24} - A_{45} - A_{51}$, length: 3, value: $0.25 \times 5 \times 0.5 = 0.625$

Accordingly, A_{21} is: $\sqrt[5]{0.2 \times 0.5 \times 0.2 \times 0.25 \times 0.625} = 0.315$

In the example above, all paths having length less than or equal to three were found for two missing entries, since the matrix size is five and missing entries are six there is small number of paths. However, as the number of alternatives increases, the number of paths

will increase. For example, in a 25×25 matrix with having two missing entries, the total number of paths that reach to these two missing entries having two and three lengths are 46 (23×2) and 1012 $(23 \times 22 \times 2)$ respectively. Moreover, in a 25×25 matrix there are paths with having from 2 to 24 arcs for these missing entries. Note that the above example is only restricted to two missing entries, the total number of paths that need to be derived will also increase as the number of missing entries rises. In order to avoid this complexity, instead of looking at all paths, this study suggests examining paths that contain a limited number "k" of arcs. Choosing "k" means, not only evaluating path lengths of "k", but also evaluating paths with lengths of less than "k". For example, k = 4 suggests taking into account the paths of having lengths of 2, 3, 4 (0 is trivial and 1 means there is a direct path from i to j, in this case A_{ij} is not a missing value). The motivations behind considering small values are time complexity and the belief that the information is stored on paths that have few arcs sufficient to approach the true value. As the length of a path increases, the opportunity of forming path increases since more combinations can be achieved by adding indirect relations. Therefore, choosing small "k" decreases the complexity of the algorithm. Furthermore, the chance of revealing less perturbation may increase, since perturbations accumulate while multiplying the arcs. However, there is a tradeoff which is the chance of information being lost. It could be that the most accurate information is stored on the longest path. In the ideal case when there is no perturbation, paths which have length of two are enough to derive the missing values correctly, considering the graph is connected.

Moreover, in this study the importance levels of the paths are investigated. As the length of the arcs increases, the number of interactions increases. This may cause the possibility of an increase in deformation. Therefore, while taking the geometric average, paths having different lengths were weighted.

The consistency of the algorithms in the AHP literature is a popular debate subject. There are several methodologies to improve the consistency of the algorithms in literature such as Cao et al. (2008) and Gomez-Ruiz et al. (2010). The methodology was proposed by Gomez-Ruiz et al. (2010) modifies the highest perturbation between pairwise comparison matrix and theoretical pairwise comparison matrix. The relation between perturbation and the matrices is as below.

$$A = W \circ E \tag{4.1}$$

where *E* represents the perturbation matrix. The idea behind the algorithm is changing the highest perturbation, so that it develops the consistency of the algorithm in a better way. They proposed to change the entry which is farthest from one. Their algorithm is demonstrated as follows (Algorithm 5).

Algorithm 5: Improving Consistency Ratio

1: **Input:** $A_{n\times n}$, $W_{n\times n}$

2: **Calculate:** $E_{n\times n}$

3: **Find:** E_{ij} that is farthest from one

4: **Replace:** A_{ij} and A_{ji} with 0 and the corresponding diagonal entries of i, j with 2

5: **Calculate:** new $\boldsymbol{w}$ according to updated $A_{n\times n}$

6: **Replace:** A_{ij} with $\frac{w_i}{w_j}$, A_{ji} with $\frac{w_j}{w_i}$ and the diagonal entries of i, j with 1

7: **Output:** *A*

Based on these considerations, the following heuristics namely Transitivity of Length of Two (TLT) (Algorithm 6) and Transitivity of Length of not Exceeding Three (TLET) (Algorithm 7) were proposed in this study. The length parameter "k" was selected as two in the TLT, and three in the TLET. As highlighted before, length parameter three means investigating paths both length of two and three. The importance parameter is only considered in the TLET because TLT utilizes only paths of length two. In this study, the importance coefficient of TLET was parameterized as one and two. If the parameter value is two, paths of length two are counted twice and paths of length three are counted once. However, if the parameter value is one, both are considered only once when calculating the geometric average.

Example 3: Consider the values of the above paths (Example 2), the geometric average for A_{12} must be calculated as follows if the importance level parameter is taken as two.

$$\sqrt[7]{5\times 2^2\times 5\times 4^2\times 1.6} = 3.068 \qquad (4.2)$$

The main reason for choosing the importance level close is the fact that paths of length two and paths of length three are exposed to similar perturbations, there is an additional arc which leads to extra multiplication in paths of length three.

In addition, the effect of improving the consistency method (Gomez-Ruiz et al., 2010) on the completed matrix created by the algorithms was adapted. For each algorithm three different scenarios, which are based on the number of entries that change (NEC), were evaluated. In the first scenario, the method was not utilized, no entry was changed. In the second scenario, the method iterated twice. In the third scenario, the method was executed four times. Thus, 9 (3 due to importance levels (1, 2) or path lengths parameters (2, 3) × 3 due to NEC parameters (0, 2, 4)) algorithms having different parameters were revealed. Finally, the calculated priority vector of the matrices generated by these algorithms are founded by using principal right eigenvector.

Algorithm 6: Transitivity of Length of Two (TLT)

1: **Input:** $A_{n\times n}$, NEC
2: **Calculate**: missing number of A
3: **Create:** 0 matrix $M_{n\times n}$
3: **for** i = 1, …, n **do**
4: **for** j = 1, …, n **do**
5: **If** ($A_{ij} \equiv 0$ and i < j)
6: **Find** and **Store:** all paths (length of 2) from i to j by using A
7: **If** (Paths were found)
8: **Calculate:** Geometric average of these paths
9: **Assign:** Geometric average to M_{ij} and reciprocal of it to M_{ji}
10: **Reduce:** missing number by 2
11: **Assign:** non-zero values of M to A
12: **while** (there is missing number)
13: **for** i =1, …, n **do**
14: **for** j =1, …, n **do**
15: **If** ($A_{ij} \equiv 0$ and i < j)
16: **Find** and **Store:** all paths (length of 2) from i to j by using A
17: **If** (Paths were found)
18: **Calculate:** Geometric average of these paths
19: **Assign:** Geometric average as A_{ij} and reciprocal of it as A_{ji}
20: **Reduce:** missing number by 2
21: **Iterate:** Improving Consistency Ratio (A) NEC times
22: **Derive:** principal right eigenvector of A
23: **Output:** $\boldsymbol{w}$

In the algorithm, there are two loops. The purpose of the first loop is to derive the paths that are associated with the missing entry by using only the matrix assigned by the decision maker. However, in the second loop paths are generated by utilizing the matrix assigned by the decision maker, the values extracted by the first loop and the values that are calculated during the second loop. The reason behind the existence of the second loop is in some cases the first loop is not enough to fill in all the missing values only adhering to the decision maker, even though the matrix derived by the decision maker is connected (see ex. 4). The reason is the constraint of the path lengths (in this algorithm it is two). While calculating the missing values, the primacy of the algorithm is to make a derivation only over the values assigned by the decision maker since they are assumed as expert knowledge. Therefore, solely the matrix assigned by the decision maker is used in the first loop. If there are still any missing entries after the first loop, the algorithm enters the second loop and remains in the second loop until there is no missing entry. After the missing values are calculated, the Improving Consistency Ratio algorithm is iterated NEC times. Finally, the calculated priority vector of the completed matrix is obtained by utilizing the principal right eigenvector.

Example 4: Suppose that the decision maker filled in a matrix as follows.

$$A = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

The matrix that the algorithm can complete in the first loop, although the matrix is connected, is as follows.

$$A^* = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

There are values that cannot be filled in the matrix. Therefore, there is a need for the second loop.

Algorithm 7: Transitivity of Length of not Exceeding Three (TLET)

1: **Input:** $A_{n \times n}$, NEC, importance
2: **Calculate**: missing number of *A*
3: **Create:** 0 matrix $M_{n \times n}$
3: **for** i = 1, …, n **do**
4: **for** j = 1, …, n **do**
5: **If** ($A_{ij} \equiv 0$ and i < j)
6: **Find** and **Store:** all paths (length of 2, 3) from i to j with their path length by using *A*
7: **If** (Paths were found)
8: **Calculate:** Geometric average of these paths based on path length and importance
9: **Assign:** Geometric average to M_{ij} and reciprocal of it to M_{ji}
10: **Reduce:** missing number by 2
11: **Assign:** non-zero values of *M* to *A*
12: **while** (there is missing number)
13: **for** i =1, …, n **do**
14: **for** j =1, …, n **do**
15: **If** ($A_{ij} \equiv 0$ and i < j)
16: **Find** and **Store:** all paths (length of 2, 3) from i to j with their path lengths by using *A*
17: **If** (Paths were found)
18: **Calculate:** Geometric average of these paths based on path length and importance
19: **Assign:** Geometric average as A_{ij} and reciprocal of it as A_{ji}
20: **Reduce:** missing number by 2
21: **Iterate:** Improving Consistency Ratio (*A*) NEC times
22: **Derive:** principal right eigenvector of *A*
23: **Output:** ***w***

As in the TLT algorithm, there are two loops in this algorithm and the purposes of the loops are the same, they differ only in content. Identifying paths with a length of three and consideration of the importance level in the geometric average are additional requirements in TLET. After the PCM is completed, the Improving Consistency Ratio algorithm is executed based on NEC parameter. Lastly, priority values are calculated by using the principal right eigenvector method.

In this algorithm, depending on the size of the matrix, the number of missing entries and the software used, there is a chance to obtain an undefined value while accumulating the

values of paths. If the matrix size is high and the missing numbers in the PCM are low, the algorithm finds an excessive number of paths while deriving the missing entry. Depending on the values of these paths, a high (low) number that is undefined in the software can be obtained from the multiplication of the values while calculating geometric average. A viable action in this case is to set this undefined high (low) number to the highest (lowest) valid number in the software. While performing the numerical experimental analysis in this research, this situation solely occurred in some iterations when the matrix size was 25 and the missingness ratio was 20 percent. This number was set as the highest or lowest value in the software depending on the value it has. According to the results of the experiment, no deflecting results were observed (see the result and discussion chapter, the algorithm was competitive in the case of matrix size 25 or missing ratio 0.2).

4.2. The Numerical Experimental Design

An approach was developed in order to mimic the process of a decision maker assigning pairwise comparisons. This methodology was adapted from the experimental setup developed by Ahmed and Kilic (2022) to the incomplete AHP context. The adapted methodology assumes that the decision maker uses a specific weight vector and expresses preferences through pairwise comparisons. The pairwise comparison matrix should constitute inconsistency and missing entries due to the decision maker inconsistency, limited knowledge of decision maker about some alternatives, uncertainty in the comparisons, and time constraints. To simulate this process, a numerical dataset was created. The dataset included PCMs of three matrix sizes (n = 5, 15, 25) three levels of inconsistencies (low, medium, high) and four missing ratios (0.2, 0.3, 0.4, 0.5). In sum, 36 (3 matrix sizes × 3 inconsistency levels × 4 missing ratios) numerical experimental conditions were obtained. The subsequent algorithm (Algorithm 8) was initially implemented to generate random inconsistent PCMs.

Algorithm 8: Creating random PCM

1: **Create:** random normalized vector $\boldsymbol{v}$,
random uniform perturbation coefficient **c**, and 0 matrix $A_{n \times n}$
2: **for** i =1, …, n **do**
3: **for** j =1, …, n **do**
4: **if** (i ≡ j)
5: A_{ij} = 1
6: **elif** (i < j)
7: a = $\boldsymbol{v_i}$ / $\boldsymbol{v_j}$ – **c** × $\boldsymbol{v_i}$ / $\boldsymbol{v_j}$
8: b = $\boldsymbol{v_i}$ / $\boldsymbol{v_j}$ + **c** × $\boldsymbol{v_i}$ / $\boldsymbol{v_j}$
9: A_{ij} = uniform (a, b)
10: A_{ji} = 1 / A_{ij}

Here, vector $\boldsymbol{v}$ represents the true priority vector that the decision maker should adhere to achieve the true theoretical pairwise comparison matrix. The inconsistency arises from the utilization of parameter **c** and the uniform operation. To maintain the integrity of the AHP structure, the diagonal entries are kept as 1, and the reciprocal property is preserved.

The existence of randomness leads the matrices to exhibit diverse degrees of inconsistencies. In order to quantify the inconsistency of these matrices, the Consistency Ratio (CR) was utilized. Matrices exhibiting a CR value ranging from 0 to less than 0.03 were considered as a low level of inconsistency. Matrices with a CR value between 0.03 and less than 0.06 were classified as a medium (med) level of inconsistency. Matrices with a CR value between 0.06 and less than 0.1 were taken into account as a high level of inconsistency. Matrices that surpassed 0.1 CR value were labeled inadequately consistent and thus were not used.

In order to introduce missing entries, some random entries in the generated matrix were set to zero based on the missing ratios (0.2, 0.3, 0.4 and 0.5), no changes were made to the diagonal entries. The number of missing entries were determined by the missing ratio and the multiplication of the number of entries in the upper-right triangle of the matrix, excluding the diagonal entries. In cases where the multiplication was not an integer, the value was rounded up. For instance, let matrix size be 10 and the missing ratio be 0.3.

The upper-right triangle without the diagonal entries, there are a total of 45 entries (Equation 4.3).

$$\frac{(n \times n-1)}{2} = 45 \tag{4.3}$$

Therefore, the number of missing entries was set as 14 (Equation 4.4).

$$\lceil 45 \times 0.3 \rceil = 14 \tag{4.4}$$

Subsequently, any entries that had set to zero were determined, and its reciprocal entry were also set to zero. Connectedness of the incomplete pairwise comparison matrix was an assumption in some methodologies that were utilized in this study. Therefore, the connectedness of the newly generated incomplete matrix was also examined. If it was connected, the methodologies computed the priority vector and were assessed according to the metrics. In case the matrix was not connected, a new random incomplete pairwise comparison matrix was created until it was connected.

For each numerical experimental condition, a total of 100 connected incomplete pairwise comparison matrices were produced and analyzed. As a result, 3600 (36 experimental conditions × 100 matrices) connected incomplete pairwise comparison matrices were obtained.

4.3. The Empirical Experimental Design

An empirical experiment including the participation of students from Sabancı University, Istanbul was conducted. In this experiment, 30 students were asked to compare the geographic size of 15 countries and to fill two matrices. In the first matrix, the students were expected to leave cells empty for which they were uncertain or preferred not to fill in. In the second matrix, they were asked to fill in the matrix completely referring to the cells they had filled in the first matrix. Participants were required to adhere the Saaty scale (Table 1) while they were assigning the cells. The priority vectors of the first matrices were determined utilizing the incomplete AHP methodologies. However, the priority vectors of the second matrices were identified using the principal right eigenvector method (Saaty, 1977). The comparison of the incomplete AHP methodologies among themselves was made using the calculated priority vectors of the first matrices and true priority vectors. The accuracy of filling the uncertain comparisons was examined by comparing the calculated priority vectors of both the Saaty methodology and the

incomplete AHP methodologies with the true priority vectors. The discussion was whether it is more suitable to leave the uncertain comparisons for the incomplete AHP methods to fill or to let the decision maker estimate them.

It is necessary that the decision maker have relevant knowledge about the subject in AHP. The reasons for choosing the geographic size of countries as the subject were that they are a kind of cultural knowledge therefore likely to be known, and they have a natural scale. The fact that they have quantitative values enabled the identification of the true priority vector, as shown below (Table 3). Thus, the priority vectors computed by the methodologies could be evaluated and compared.

Table 3 Normalized true priority vector of empirical experiment

Countries	**Corresponding Priority Vector Value**
Chile	0.1
Colombia	0.151
Egypt	0.133
France	0.073
Germany	0.047
Greece	0.017
Japan	0.05
Morocco	0.059
Portugal	0.012
Senegal	0.026
Spain	0.067
Sweden	0.06
Thailand	0.068
Turkey	0.104
United Kingdom	0.032

The experiment was approved by the Sabancı University ethics committee. Moreover, before it was conducted by participants, the written informed consent form had been taken from the participants. The estimated duration of the experiment was approximately 10 minutes. However, additional time was granted to the participants who asked.

5. RESULTS AND DISCUSSIONS

5.1. Numerical Experimental Design

The existing six algorithms in literature, namely Harker, Dematel, Sparse Eigenvector, Metropolis-Hastings, Heat-Bath, and SLLS, along with the parametric heuristics TLT and TLET algorithms, were compared in terms of six performance metrics, including CI, Kendall's tau, MCIV, Euclidean distance, GCIV-VW and central processing unit time. The TLT algorithm was examined in three different ways according to the NEC parameter (NEC = 0, NEC = 2, NEC = 4). The TLET algorithm was analyzed in six different ways regarding the NEC and importance parameters (three different NEC values = 0, 2, 4, and two different importance values = 1, 2).

Before comparing results of the methodologies, the relationships between performance metrics were examined in order to reduce the complexity since there are six different metrics and to see the behavior between the metrics. For this purpose, the correlation coefficient was utilized. Data was collected for each experimental condition from each algorithm. In order to avoid oversampling of the proposed heuristics, only one TLT algorithm and one TLET algorithm were selected from the three TLT algorithms and six TLET algorithms, namely TLT - NEC = 2 and TLET - NEC = 2, importance = 2. This selection was randomly made.

In order to observe the relationships among the metrics, data generated from numerical experimental setup were gathered. Five metrics out of six metrics (CI, Kendall's tau, MCIV, Euclidean distance, and GCIV-VW) measure the accuracy and deviation of the algorithms' performance, while one metric (central processing unit time) examines the algorithm's process time. Therefore, the relationships among only the five metrics, which

are related with accuracy and deviation, were investigated by using correlation coefficients.

Since four algorithms namely Sparse Eigenvector, Metropolis-Hastings, Heat-Bath, and SLLS had a sparse setting, the CI values could not be derived. These values had been assigned zero during the numerical experimental setup. Consequently, when examining the relationships between the CI and the other four metrics, the data from these four algorithms could not be used. The analysis for CI was conducted only based on data collected from the Harker, the Dematel, the TLT - NEC = 2, and the TLET - NEC = 2, importance = 2 algorithms.

The correlation coefficient values were interpreted as follows: (0, 0.3) or (-0.3, 0) indicate a weak, (0.3, 0.7) or (-0.3, -0.7) define a moderate, and (0.7, 1) or (-0.7, -1) specify a strong linear relationship (Ratner, 2009). The relationships between metrics are shown in table 4, and the strong linear relationships are demonstrated in figures 1 and 2. Strong linear relationships were highlighted in bold.

Table 4 Correlation coefficients between metrics

Metrics	Correlation Coefficient
Kendall's tau – GCIV-VW	-0.511
Kendall's tau – MCIV	-0.074
Kendall's tau – Euclidean distance	-0.576
Kendall's tau – CI	-0.012
GCIV-VW – MCIV	0.001
GCIV-VW – Euclidean distance	**0.857**
GCIV-VW – CI	-0.078
Euclidean distance – MCIV	-0.036
Euclidean distance – CI	-0.126
MCIV – CI	**0.87**

Figure 1 Strong linear relationship between GCIV-VW and Euclidean distance

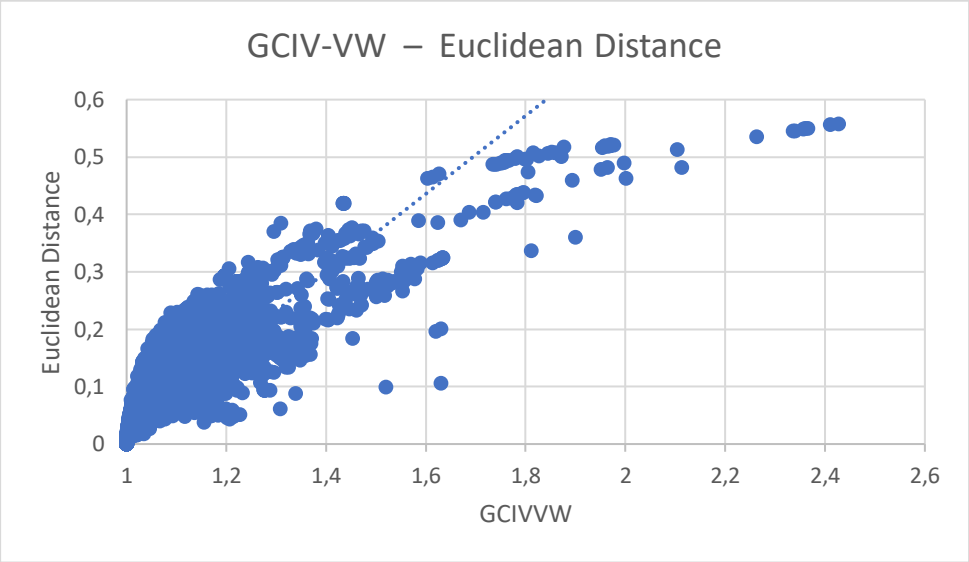
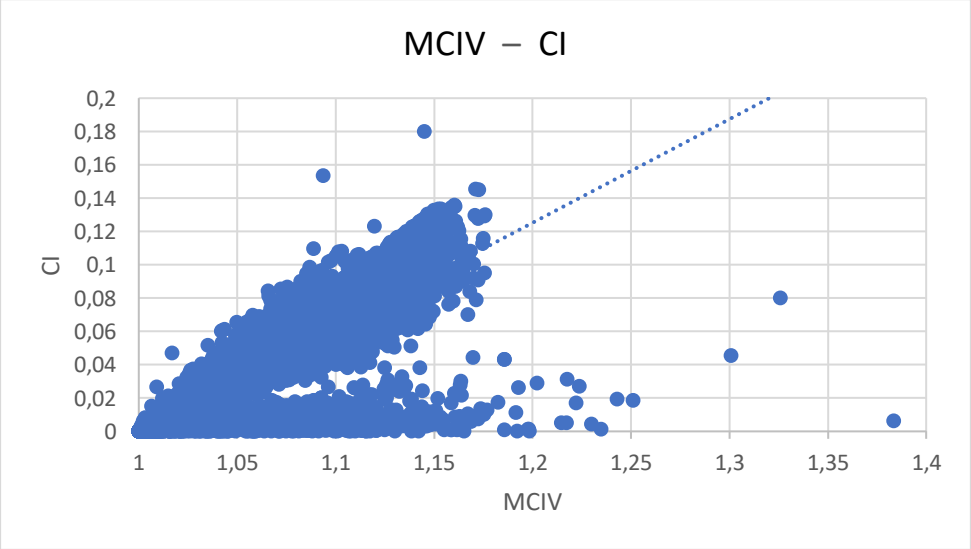


Figure 2 Strong linear relationship between MCIV and CI



The correlation coefficient between GCIV-VW and Euclidean distance was 0.857, and the correlation coefficient between MCIV and CI was 0.87, which proposed a strong linear relationship. Therefore, it was assumed that Euclidean distance can be interpreted with GCIV-VW, as well as CI with MCIV. Note that the relationships were investigated as only linearly, there may exist another type of relations. This is out of the scope of this research. However, it was noted as a future study subject. In this study, the aim was to reduce the complexity caused by metric numbers in a meaningful way. Therefore, only two out of these four metrics were used, while comparing the algorithms. Another contribution of this result rather than reducing the metric number, is the advantage of

being able to make interpretations about the CI values to some degree. As mentioned earlier, the CI values for the four algorithms could not be extracted because of their sparse setting. However, thanks to the relationship between MCIV and CI, it becomes possible to make inferences. Therefore, MCIV was chosen among the two. GCIV-VW was chosen instead of Euclidean distance since it was created in the context of AHP. In short, the algorithms outputs were decided to compare using four metrics instead of six, these were GCIV-VW, MCIV, Kendall's tau, and central processing unit time.

In order to reduce complexity comes from the algorithm number before comparing all algorithms, the parametric algorithms were grouped, and the algorithm that would yield the best statistical results based on the metrics within each group was distinguished. Therefore, one TLT algorithm was chosen from the three TLT algorithms, and one TLET algorithm was selected from the six TLET algorithms. Subsequently, a total of eight algorithms (the two proposed algorithms and the six algorithms from the literature) were compared according to the metrics.

Tukey HSD and Games Howell tests were utilized as a statistical comparison test. The Tukey HSD test assumes samples have equal variances (Lee and Lee, 2018). It was employed when the GCIV-VW, MCIV, and Kendall's tau metrics were compared, since the algorithms outputs had equal or close variances. On the other hand, the Games Howell test, which can be used when inequality of variances existed (Lee and Lee, 2018), was used to compare the central processing unit time where were significant deviations on variances among the outputs of the algorithms. Statistical differences were examined at the significance level of 0.05.

In the experimental conditions, there were three parameters including matrix size, missing ratio, and consistency. The 36 experimental conditions result were grouped into 10 cases for each metrics these are high consistency, med consistency, low consistency, matrix size 5, matrix size 15, matrix size 25, missing ratio 0.2, missing ratio 0.3, missing ratio 0.4, and missing ratio 0.5. While creating the cases, one parameter of experimental conditions was kept constant, and the data were collected accordingly. For example, in the matrix size 5 case, considering three consistency levels and four missing ratios, a total of 1200 ($3 \times 4 \times 100$ simulations/experimental design) data were gathered. Therefore, there were 1200 ($3 \times 4 \times 100$) data in the consistency and matrix size cases, while there were 900 ($3 \times 3 \times 100$) data in the missing ratio cases.

5.1.1. TLT Comparisons

The TLT algorithms (NEC = 0, NEC = 2, NEC = 4) were compared statistically based on three metrics namely GCIV-VW, MCIV, and Kendall's tau. Since these algorithms displayed similar processing time values, their comparison based on central processing unit time was omitted. The following tables (Table 5, 6 and 7) illustrate the comparisons of these three algorithms for a total of 30 cases (10 cases for each three metrics). Entries demonstrating statistical disparities were highlighted in bold.

Table 5 Results of the Tukey HSD test for matrix size cases

Metric	**i**	**j**	**Mean Diff. (i-j) n=5**	**Sig.**	**Mean Diff. (i-j) n = 15**	**Sig.**	**Mean Diff. (i-j) n = 25**	**Sig.**
	NEC=0	NEC=2	-0.018	**0.003**	-0.001	0.574	-0.001	0.961
GCIV-VW	NEC=0	NEC=4	-0.023	**<0.001**	-0.001	0.179	-0.001	0.886
	NEC=2	NEC=4	-0.006	0.941	-0.000	0.999	-0.000	1.000
	NEC=0	NEC=2	-0.020	**<0.001**	-0.002	0.976	-0.001	1.000
MCIV	NEC=0	NEC=4	-0.025	**<0.001**	-0.003	0.656	-0.001	1.000
	NEC=2	NEC=4	-0.005	**0.020**	-0.001	0.998	-0.000	1.000
	NEC=0	NEC=2	0.019	0.522	0.004	0.916	0.002	0.992
Kendall	NEC=0	NEC=4	0.025	0.176	0.005	0.859	0.002	0.976
	NEC=2	NEC=4	0.006	1.000	0.000	1.000	0.000	1.000

Table 6 Results of the Tukey HSD test for consistency cases

Metric	**i**	**j**	**Mean Diff. (i-j) Low**	**Sig.**	**Mean Diff. (i-j) Med**	**Sig.**	**Mean Diff. (i-j) High**	**Sig.**
	NEC=0	NEC=2	-0.002	0.310	-0.007	**0.002**	-0.010	0.269
GCIV-VW	NEC=0	NEC=4	-0.002	**0.047**	-0.010	**<0.001**	-0.014	**0.033**
	NEC=2	NEC=4	-0.001	0.998	-0.002	0.959	-0.003	0.997
	NEC=0	NEC=2	-0.002	**0.026**	-0.007	**<0.001**	-0.013	**<0.001**
MCIV	NEC=0	NEC=4	-0.002	**0.001**	-0.009	**<0.001**	-0.018	**<0.001**
	NEC=2	NEC=4	-0.000	0.995	-0.002	0.278	-0.004	0.228
	NEC=0	NEC=2	0.005	0.918	0.009	0.860	0.011	0.830
Kendall	NEC=0	NEC=4	0.007	0.724	0.014	0.318	0.011	0.838
	NEC=2	NEC=4	0.002	1.000	0.005	0.995	-0.000	1.000

Table 7 Results of the Tukey HSD test for missing ratio cases

Metric	i	j	Mean (i-j) 0.2	Sig.	Mean (i-j) 0.3	Sig.	Mean (i-j) 0.4	Sig.	Mean (i-j) 0.5	Sig.
	NEC=0	NEC=2	-0.007	0.110	-0.007	0.258	-0.007	0.471	-0.005	0.950
GCIV-VW	NEC=0	NEC=4	-0.008	**0.028**	-0.008	0.060	-0.010	0.081	-0.008	0.634
	NEC=2	NEC=4	-0.001	1.000	-0.002	1.000	-0.003	0.996	-0.003	0.999
	NEC=0	NEC=2	-0.007	**0.016**	-0.008	**0.006**	-0.008	**0.002**	-0.006	0.157
MCIV	NEC=0	NEC=4	-0.009	**<0.001**	-0.010	**<0.001**	-0.012	**<0.001**	-0.008	**0.003**
	NEC=2	NEC=4	-0.002	0.999	-0.002	0.995	-0.003	0.792	-0.003	0.944
	NEC=0	NEC=2	0.012	0.647	0.008	0.971	0.006	0.996	0.008	0.987
Kendall	NEC=0	NEC=4	0.013	0.579	0.008	0.957	0.012	0.833	0.010	0.946
	NEC=2	NEC=4	0.001	1.000	0.001	1.000	0.005	0.998	0.002	1.000

Note that MCIV and GCIV-VW are a distance metric, while Kendall is a similarity metric. There is no significant difference among the three algorithms in terms of the Kendall metric. However, according to the GCIV-VW and MCIV metrics, there are statistically significant differences in particular cases. According to GCIV-VW, in cases including matrix size 5, low consistency, med consistency, high consistency, and missing ratio 0.2 statistically differences were observed. According to MCIV, in cases namely matrix size 5, low consistency, med consistency, high consistency, missing ratio 0.2, missing ratio 0.3, missing ratio 0.4, and missing ratio 0.5 statistically differences were obtained. NEC = 0 outperforms the other two algorithms in these cases. Furthermore, no statistical difference was observed between NEC = 2 and NEC = 4, except the case of matrix size 5 based on MCIV. In that case NEC = 2 displays statistically superiority over NEC = 4. The fact that the NEC = 0 algorithm outperforms the other two algorithms demonstrates that the increase in the NEC parameter is not beneficial in terms of the MCIV and GCIV-VW metrics in the TLT algorithm. Therefore, among the TLT algorithms NEC = 0 was selected.

5.1.2. TLET Comparisons

The statistical comparison of the six TLET algorithms were conducted based on three metrics, which are GCIV-VW, MCIV, and Kendall's tau. Since the processing time values of the algorithms were similar, their statistical comparison based on central processing unit time was not investigated. Comparisons were made among six algorithms, so that

resulted in 15 different comparisons for each 30 cases. Due to limited space and for the sake of compactness, only cases where there was at least one statistically significant difference between any of the algorithms were presented in tables below (Table 8, 9 and 10) and in the appendix chapter (Table 27, 28, 29, 30, 31 and 32). The representation in the tables is as follows: The case title is indicated above the tables. Mean differences between algorithms and corresponding significance values are displayed within the cells. The upper value in the cell corresponds to the mean difference, and the lower value indicates the significance value. The mean difference was obtained by subtracting the algorithm in the column from the algorithm in the row. Only the right upper triangle was demonstrated, the bottom left triangle was not presented since the left bottom triangle is negative of the right upper triangle in terms of mean differences, and the same in terms of the significance values. The diagonal entries were also not indicated as was considered trivial. Entries that show statistical differences were highlighted in bold.

Note 3: "importance" was abbreviated as "imp" in the tables.

Table 8 Results of the Tukey HSD test for missing ratio 0.2 case based on GCIV-VW

	NEC = 2, imp = 1	NEC = 4, imp = 1	NEC = 0, imp = 2	NEC = 2, imp = 2	NEC = 4, imp = 2
NEC = 0, imp = 1	-0.007 0.170	-0.008 0.052	0 1.00	-0.006 0.23	-0.007 0.085
NEC = 2, imp = 1		-0.001 1.00	0.007 0.165	0 1.00	-0.001 1.00
NEC = 4, imp = 1			0.008 **0.05**	0.001 1.00	0 1.00
NEC = 0, imp = 2				-0.006 0.224	-0.007 0.082
NEC = 2, imp = 2					-0.001 1.00

Table 9 Results of the Tukey HSD test for matrix size 5 case based on MCIV

	NEC = 2, imp = 1	NEC = 4, imp = 1	NEC = 0, imp = 2	NEC = 2, imp = 2	NEC = 4, imp = 2
NEC = 0, imp = 1	-0.014 **<0.001**	-0.018 **<0.001**	0 1.00	-0.012 **<0.001**	-0.014 **<0.001**
NEC = 2, imp = 1		-0.004 0.165	0.014 **<0.001**	0.002 0.91	-0 1.00
NEC = 4, imp = 1			0.018 **<0.001**	0.006 **0.002**	0.004 0.222
NEC = 0, imp = 2				-0.012 **<0.001**	-0.014 **<0.001**
NEC = 2, imp = 2					-0.002 0.856

Table 10 Results of the Tukey HSD test for missing ratio 0.2 case based on MCIV

	NEC = 2, imp = 1	NEC = 4, imp = 1	NEC = 0, imp = 2	NEC = 2, imp = 2	NEC = 4, imp = 2
NEC = 0, imp = 1	-0.001 **0.033**	-0.008 **0.002**	-0 1.00	-0.007 **0.04**	-0.008 **0.003**
NEC = 2, imp = 1		-0.002 0.998	0.007 **0.034**	0 1.00	-0.001 0.999
NEC = 4, imp = 1			0.008 **0.002**	0.002 0.997	0 1.00
NEC = 0, imp = 2				-0.007 **0.04**	-0.008 **0.003**
NEC = 2, imp = 2					-0.002 0.998

When examining the obtained results, there are no significant differences among the algorithms in terms of the Kendall metric. However, significant differences appeared between the algorithms in particular cases. Based on the GCIV-VW metric, significant differences were obtained in the cases of matrix size 5 (Table 27), med consistency (Table 28), and missing ratio 0.2 (Table 8). In cases matrix size 5 and med consistency, the algorithms NEC = 0, importance = 1 and NEC = 0, importance = 2 demonstrated significantly better results compared to other algorithms. However, for the missing ratio 0.2 case, only NEC = 0, importance = 2 dominated statistically over another algorithm. In terms of the MCIV metric, significant differences emerged in the cases of matrix size 5 (Table 9), med consistency (Table 29), high consistency (Table 30), missing ratio 0.2 (Table 10), missing ratio 0.3 (Table 31), and missing ratio 0.5 (Table 32). The algorithms NEC = 0, importance = 1 and NEC = 0, importance = 2 displayed significantly better results in these cases.

Nine cases out of 30 indicated statistically significant differences. In these cases, generally, the algorithms NEC = 0, importance = 1 and NEC = 0, importance = 2 showed better performance. They do not create a statistically significant difference between each other. However, in one case only the NEC = 0, importance = 2 algorithm displayed superiority over the NEC = 4, importance = 1 algorithm. Moreover, the importance level of the dominated algorithm was 1. Therefore, the NEC = 0, importance = 2 algorithm was selected among the TLET algorithms.

The algorithms with having NEC parameters as zero displayed statistically superiority over the algorithms with having NEC parameters as two and four in the cases where statistically significance difference exists. This result might propose that a rise in the NEC

value diminishes the MCIV and GCIV-VW performances. However, depending only on the importance values there was not any dominance over the algorithms. Although the importance parameter does not severely affect these metrics, its variation might create differences under certain cases (e.g., missing ratio 0.2 case based on GCIV-VW).

5.1.3. All Algorithms Comparisons

Eight algorithms, namely Harker, Dematel, Sparse Eigenvector, Metropolis-Hastings, Heat-Bath, SLLS, TLT-NEC=0, and TLET-NEC=0, importance = 2 were statistically compared based on four metrics, including GCIV-VW, MCIV, Kendall's tau, and central processing unit time. 40 cases resulted in 28 different comparisons among the eight algorithms were considered. Due to limited space and conciseness, only the cases where there was a statistical difference between any of the algorithms were included in the tables. Initially, the tables corresponding to GCIV-VW, MCIV, and Kendall's tau metrics were presented and discussed below (Table 11, 12, and 13) and in the appendix (Table 33 and 34) for comparing the accuracy of the algorithms. While comparing the accuracy, comparisons of the algorithms were investigated extensively since the results had been close. Therefore, some additional tables were indicated after statistical comparisons based on these 3 metrics. Subsequently, the tables associated with central processing unit time were revealed and discussed in order to compare computation time performance. The representation and purpose of the tables align with the TLET Comparisons section.

Table 11 Results of the Tukey HSD test for matrix size 25 case based on GCIV-VW

	Metropolis Hastings	Heat Bath	SLLS	Harker	Dematel	TLT NEC = 0	TLET NEC = 0, imp = 2
Sparse Eigenvector	-0.001 0.871	0 0.996	0.001 0.504	-0 0.999	0 0.999	0 0.999	0 0.996
Metropolis Hastings		0.001 0.414	0.002 **0.019**	0 0.995	0.001 0.531	0.001 0.513	0.001 0.403
Heat Bath			0.001 0.921	-0.001 0.88	-0 1.00	-0 1.00	0 1.00
SLLS				-0.001 0.161	-0.001 0.854	-0.001 0.865	-0.001 0.926
Harker					0.001 0.939	0.001 0.932	0.001 0.873
Dematel						0 1.00	0 1.00
TLT-NEC=0							0 1.00

Table 12 Results of the Tukey HSD Test for med consistency case based on MCIV

	Metropolis Hastings	Heat Bath	SLLS	Harker	Dematel	TLT NEC = 0	TLET NEC = 0, imp = 2
Sparse Eigenvector	-0.003 0.07	-0.001 0.863	0 1.00	-0 1.00	-0 1.00	-0 1.00	0 1.00
Metropolis Hastings		0.001 0.795	0.003 **0.017**	0.003 0.108	0.003 0.083	0.002 0.169	0.003 **0.048**
Heat Bath			0.002 0.584	0.001 0.923	0.001 0.888	0.001 0.968	0.001 0.794
SLLS				-0.001 0.999	-0 1.00	-0.001 0.993	-0 1.00
Harker					0 1.00	-0 1.00	0 1.00
Dematel						-0 1.00	0 1.00
TLT-NEC=0							0 1.00

Table 13 Results of the Tukey HSD Test for high consistency case based on MCIV

	Metropolis Hastings	Heat Bath	SLLS	Harker	Dematel	TLT NEC = 0	TLET NEC = 0, imp = 2
Sparse Eigenvector	-0.006 **0.028**	-0.002 0.931	0.001 0.993	-0 1.00	0 1.00	-0 1.00	0.001 1.00
Metropolis Hastings		0.003 0.472	0.007 **0.001**	0.005 0.063	0.006 **0.014**	0.005 **0.036**	0.006 **0.009**
Heat Bath			0.003 0.476	0.002 0.983	0.002 0.85	0.002 0.952	0.003 0.783
SLLS				-0.002 0.963	-0.001 0.999	-0.002 0.987	-0.001 1.00
Harker					0.001 1.00	0 1.00	0.001 0.999
Dematel						-0.001 1.00	0 1.00
TLT-NEC=0							0.001 1.00

Statistically significant differences were not observed in most cases. Out of a total of 30 cases, only in five cases (Table 11, 12, 13, 33 and 34) at least one statistically significant difference was acquired. The cases where statistically significant differences were obtained are as follows, matrix size 25 based on GCIV-VW (Table 11), med consistency based on MCIV (Table 12), high consistency based on MCIV (Table 13), matrix size 5 based on MCIV (Table 34), and matrix size 25 based on Kendall (Table 33). Even though statistically significant differences were obtained in these cases, the occurrences of these differences are limited. In the cases of matrix size 25 based on Kendall, matrix size 5

based on MCIV, and matrix size 25 based on GCIV-VW, the only statistically significant difference acquired was that the SLLS algorithm outperforms the Metropolis Hastings algorithm. In the case of med consistency based on MCIV, the SLLS and the TLET-NEC = 0, importance = 2 algorithms dominate the Metropolis Hastings algorithm. Lastly, in the case of high consistency based on MCIV, the Sparse Eigenvector, the SLLS, the Dematel, the TLT-NEC = 0, and the TLET-NEC = 0, importance = 2 algorithms have superiority over the Metropolis Hastings algorithm. The algorithms achieved similar statistical outcomes in three performance metrics in most of the cases. The Metropolis Hastings algorithm is dominated in cases where statistical differences were observed, while the SLLS algorithm dominates. Moreover, the second algorithm that has most superiority over the Metropolis Hastings algorithm is the TLET-NEC = 0, importance = 2 algorithm.

In addition to these statistical significance tests the algorithms' average performances were ranked based on the Kendall, GCIV-VW, and MCIV metrics for each 30 cases. The algorithms displaying the best and worst two average performances in each case were demonstrated in the tables below (14, 15 and 16) and in the appendix chapter (35, 36 and 37). "x" in the table shows that the algorithm in corresponding case is either one of the two best performing algorithms or one of the two worst-performing algorithms. The algorithms were indicated in rows, while the cases were shown in columns.

Note 4: In the tables the abbreviations are as follows, “N” for matrix size, “MR” for missing ratio, and “Con” for consistency.

Table 14 The best two algorithms in terms of mean based on GCIV-VW

	N 5	N 15	N 25	MR 0.2	MR 0.3	MR 0.4	MR 0.5	Low Con	Med Con	High Con
Sparse Eigenvector	X									
Metropolis Hastings										
Heat Bath							X			
SLLS		X	X	X	X	X	X	X	X	X
Harker										
Dematel	X				X	X				X
TLT-NEC = 0										
TLET NEC = 0, imp = 2		X	X	X				X	X	

Table 15 The best two algorithms in terms of mean based on MCIV

	N 5	N 15	N 25	MR 0.2	MR 0.3	MR 0.4	MR 0.5	Low Con	Med Con	High Con
Sparse Eigenvector								X		
Metropolis Hastings										
Heat Bath										
SLLS	X	X	X	X	X	X	X	X	X	X
Harker										
Dematel				X						
TLT-NEC = 0										
TLET NEC = 0, imp = 2	X	X	X		X	X	X		X	X

Table 16 The worst two algorithms in terms of mean based on GCIV-VW

	N 5	N 15	N 25	MR 0.2	MR 0.3	MR 0.4	MR 0.5	Low Con	Med Con	High Con
Sparse Eigenvector										
Metropolis Hastings	X	X	X	X	X	X		X	X	X
Heat Bath	X			X		X		X	X	
SLLS										
Harker		X	X		X		X			X
Dematel										
TLT-NEC = 0							X			
TLET NEC = 0, imp = 2										

Based on the tables indicating the best two algorithms (Table 14, 15 and 35), the SLLS algorithm appears most frequently, followed by the TLET-NEC = 0, importance = 2 algorithm. These two algorithms were also the most two algorithms that have statistically significant dominance over. Moreover, these two algorithms were not observed as the worst two algorithms in terms of average performance in any case. Based on the tables demonstrating the two worst algorithms (Table 16, 36 and 37), the Metropolis Hastings and the Heat Bath algorithms are displayed generally as the two worst algorithms in terms of average performance based on these metrics. Among these two algorithms, the Heat Bath algorithm is seen once as among two best algorithms (missing ratio 0.5 based on GCIV-VW (Table 14)), while the Metropolis Hastings algorithm could not be observed as one of the best two algorithms in any case.

There is diversity in the algorithms exhibiting the best/worst two average performances on a case basis. However, this diversity is limited. Certain algorithms, namely SLLS and TLET-NEC = 0, importance = 2, Metropolis Hastings and Heat Bath were the best/worst two performances on average in a considerable number of cases.

SLLS and TLET-NEC = 0, importance = 2 algorithms, which have most frequent demonstration among the best two performances on average basis across these three

metrics, were analyzed further to identify their superiority over each other on an instance basis. The number of instances where one algorithm outperformed the other per case was determined. The results were indicated in the following tables (Table 17, 18 and 19). The numbers in the tables display the quantity of superiority instance over the other algorithm in case basis. The tables do not show instances where there is equality, but they can be extracted by subtracting the column sum from the total number of simulations.

Note that in missing ratio cases there are 900 instances, in matrix size and consistency cases there are 1200 instances.

Table 17 The number of instances that algorithms have better performance in each case based on GCIV-VW

GCIV-VW	N 5	N 15	N 25	MR 0.2	MR 0.3	MR 0.4	MR 0.5	Low Con	Med Con	High Con
SLLS	623	680	764	524	499	517	527	651	660	756
TLET NEC = 0, imp = 2	562	520	436	376	401	383	358	543	539	436

Table 18 The number of instances that algorithms have better performance in each case based on MCIV

MCIV	N 5	N 15	N 25	MR 0.2	MR 0.3	MR 0.4	MR 0.5	Low Con	Med Con	High Con
SLLS	1176	1200	1200	900	900	900	876	1192	1197	1187
TLET NEC = 0, imp = 2	1	0	0	0	0	0	1	1	3	0

Table 19 The number of instances that algorithms have better performance in each case based on Kendall

Kendall	N 5	N 15	N 25	MR 0.2	MR 0.3	MR 0.4	MR 0.5	Low Con	Med Con	High Con
SLLS	33	268	370	160	161	178	172	52	248	371
TLET NEC = 0, imp = 2	21	186	220	89	105	110	123	25	189	213

The MCIV outcomes were as expected because of the objective function utilized in SLLS. The SLLS algorithm objective aims to minimize the logarithmic quadratic error between the pairwise comparison matrix derived from the decision maker (*A*) and the theoretical pairwise comparison matrix (*W*) and MCIV reveals the disparity between *A* and *W*. Nevertheless, it is important to remember that the SLLS algorithm was not able to establish statistically significant dominance over the TLET-NEC = 0, importance = 2 algorithm in any case. Moreover, according to Kendall and GCIV-VW results the SLLS algorithm has more competitive performance than the TLET-NEC = 0, importance = 2 algorithm. However, the TLET algorithm still provides a considerable number of advantages. Therefore, there might be a reason for choosing TLET-NEC = 0, importance = 2 in some conditions.

The statistically significance test results in the 10 cases according to the algorithms computation time performance are shown below (Table 20, 21 and 22) and in the appendix (38, 39, 40, 41, 42, 43 and 44). Note that the differences are provided in units of seconds.

Table 20 Results of the Games Howell test for matrix size 5 case based on central processing unit time

	Metropolis Hastings	Heat Bath	SLLS	Harker	Dematel	TLT NEC = 0	TLET NEC = 0, imp = 2
Sparse Eigenvector	-0.005 <**0.001**	-0.008 <**0.001**	-0.019 <**0.001**	0 0.997	-0 0.994	0 0.194	0 0.611
Metropolis Hastings		-0.003 0.136	-0.014 <**0.001**	0.005 <**0.001**	0.005 <**0.001**	0.005 <**0.001**	0.005 <**0.001**
Heat Bath			-0.011 **0.001**	0.008 <**0.001**	0.008 <**0.001**	0.008 <**0.001**	0.008 <**0.001**
SLLS				0.019 <**0.001**	0.019 <**0.001**	0.019 <**0.001**	0.019 <**0.001**
Harker					-0 0.825	0 0.55	0 0.944
Dematel						0.0002 **0.033**	0 0.181
TLT-NEC=0							-0 0.992

Table 21 Results of the Games Howell test for matrix size 25 case based on central processing unit time

	Metropolis Hastings	Heat Bath	SLLS	Harker	Dematel	TLT NEC = 0	TLET NEC = 0, imp = 2
Sparse Eigenvector	-0.042 <**0.001**	-0.079 <**0.001**	-0.18 <**0.001**	-0 0.478	-0 1.00	-0.002 <**0.001**	-0.031 <**0.001**
Metropolis Hastings		-0.037 <**0.001**	-0.138 <**0.001**	0.041 <**0.001**	0.042 <**0.001**	0.04 <**0.001**	0.011 0.21
Heat Bath			-0.101 <**0.001**	0.078 <**0.001**	0.079 <**0.001**	0.077 <**0.001**	0.048 <**0.001**
SLLS				0.18 <**0.001**	0.18 <**0.001**	0.179 <**0.001**	0.149 <**0.001**
Harker					0 0.537	-0.001 <**0.001**	-0.03 <**0.001**
Dematel						-0.002 <**0.001**	-0.031 <**0.001**
TLT-NEC=0							-0.029 <**0.001**

Table 22 Results of the Games Howell test for high consistency case based on central processing unit time

	Metropolis Hastings	Heat Bath	SLLS	Harker	Dematel	TLT NEC = 0	TLET NEC = 0, imp = 2
Sparse Eigenvector	-0.022 <**0.001**	-0.046 <**0.001**	-0.071 <**0.001**	-0 0.88	0 0.997	-0.001 **0.002**	-0.011 <**0.001**
Metropolis Hastings		-0.023 **0.028**	-0.049 <**0.001**	0.022 <**0.001**	0.022 <**0.001**	0.022 <**0.001**	0.011 **0.026**
Heat Bath			-0.025 0.413	0.045 <**0.001**	0.046 <**0.001**	0.045 <**0.001**	0.034 <**0.001**
SLLS				0.071 <**0.001**	0.071 <**0.001**	0.07 <**0.001**	0.06 <**0.001**
Harker					0 0.461	-0 0.125	-0.011 <**0.001**
Dematel						-0.001 <**0.001**	-0.011 <**0.001**
TLT-NEC=0							-0.011 <**0.001**

Based on central processing unit time, the SLSS algorithm was dominated by all other algorithms in 5 cases (Table 20, 21, 39, 40 and 44). Other 5 cases it was dominated by all except the Heat Bath algorithm (Table 22, 38, 41, 42 and 43). The Heat Bath and Metropolis Hastings algorithms generally displayed inferior performance compared to algorithms other than SLSS. The inferior performance of the SLSS, the Heat Bath and the Metropolis Hastings algorithms might be related to their convergence (termination)

parameters selected by this research. However, for precise convergence, the termination parameter must be strict, which may increase the converge time of the algorithms. The Sparse Eigenvector, the Harker and the Dematel algorithms generally had superiority. However, the TLT and TLET algorithms demonstrated their superiority on a case basis depending on the number of paths found by these algorithms. For example, in the case of matrix size 5, where the number of possible paths is low (Table 20), the TLT algorithm outperformed the Dematel algorithm, while there is no statistically significant difference observed with the Sparse Eigenvector and the Harker algorithms. Furthermore, there is no statistically significant difference between the TLET algorithm and the Dematel, the Harker, and the Sparse Eigenvector algorithms in this case. However, in the case of matrix size 25, where the number of possible paths is high (Table 21), the Dematel, the Harker, and the Sparse Eigenvector algorithms outperformed the TLT and the TLET algorithms.

Although there are statistically significant differences between algorithms, the central processing unit time of the algorithms are generally within acceptable levels. For example, in the matrix size 25 case the SLSS algorithm instance having a maximum computation time had 10.48 seconds computation time. However, these numbers may become significant when increasing the number of instances or dealing with more extreme cases.

Note that central processing unit time is highly dependent on how algorithms are coded. An expert in coding can achieve better computation time. Consequently, the mean differences between algorithms may alter. In this study, objectivity was tried to be ensured by coding the algorithms by a single reference person.

5.2. Empirical Experimental Design

Numerical studies enable us to generate different experimental conditions and to compare algorithms' performances. Compared to empirical studies, in numerical studies data might be generated more quickly and easily. However, despite efforts to reflect real eliciting processes, real-world situations include uncertainties and unknowns. Therefore,

in this study, along with numerical experimental design, comparison of algorithms was also conducted in empirical experimental design.

The data obtained from the empirical experiment, which is applied on 30 Sabancı University students, was cleaned. Although the countries' geographical sizes were selected in order to reflect expert knowledge, the students may not have had a tendency towards this cultural knowledge. Furthermore, the experiment and algorithms are dependent on principles, namely connectivity, consistency, and applicability. The incomplete methodologies include the connectivity assumption. Therefore, the first matrix that is assigned by the participants must form a connected graph to compare algorithms. In order to use the PCMs created by participants, their CR values must be less than or equal to 0.1. The participants were required to fill the cells in the matrices in a readable manner, adhering to the Saaty scale. Moreover, when completing the second matrix, it was necessary to maintain the filled cells of the first matrix for coherency.

In order to analyze consistency, the second matrices composed by the participants were utilized. The reason for using the second matrices rather than the first was completeness of the second matrices. Consistency check is important since it allows to understand the attitude exhibited by the participants in responding to the experiment and their knowledge to a degree. It was assumed that the inconsistent filling of the matrices by the participants indicated that they did not pay sufficient attention to the experiment, or they have inadequate knowledge about the subject. 8 participants out of 30 did not assign sufficient consistency. In addition to this metric, the Kendall metric was examined to assess the knowledge of the participants. Thus, one data for which the Kendall metric was calculated negatively by all the algorithms, was eliminated. Furthermore, 3 data were eliminated because of the lack of connectivity of the first matrix and 3 data were removed due to not adhering to the experimental applicability. Consequently, 15 data remained. The number of acceptances and non-acceptances along with reasons are tabulated below (Table 23).

Table 23 The number of acceptances and non-acceptances

Explanations	Amounts
Participants	30
Acceptances	15
Non-Acceptances • Consistency • Connectivity • Applicability • Kendall	15 8 3 3 1

Priority vectors of these accepted 15 data's first matrix were derived by the 15 incomplete AHP methodologies namely the existing six algorithms in the literature and the nine proposed parametric algorithms. Priority vectors of 15 data's second matrix were calculated by the Saaty eigenvector method. As a result, 16 different priority vectors were obtained for each data, and they were assessed utilizing the true priority vector based on three metrics including Kendall, GCIV-VW and MCIV.

One of the objectives of this experiment is to compare incomplete AHP methodologies. The second objective is to determine the decision maker's attitude towards uncertain pairwise comparisons. Should the decision maker predict the uncertain pairwise comparisons, or should the prediction process be left to the incomplete AHP methodologies? The decision maker's prediction process on the missing entries was simulated by asking the participants to compose the second matrix in this experiment.

Algorithm performances based on each metric are presented in the tables below (Table 24, 25 and 26). The algorithm that achieves the best outcome for each data in each metric is highlighted in bold. The mean values of the 15 performances of each algorithm according to each metric are provided in the bottom row of the corresponding tables. The algorithms that yield one of the two best mean results is emphasized in bold in this row.

Note that the MCIV metric developed for incomplete AHP methodologies. Therefore, when examining this metric, the priority vectors obtained from the second matrices using the Saaty eigenvector method were not considered. Consequently, only incomplete AHP methodologies were compared according to MCIV.

The TLT - NEC = 0 algorithm among the TLT algorithms, the TLET-NEC = 0, importance = 2 algorithm among the TLET algorithms yielded the best mean performance, like in the numerical experimental design, in terms of these three metrics. Therefore, only the TLT - NEC = 0 algorithm and TLET - NEC = 0, importance = 2 algorithm were demonstrated as TLT and TLET respectively.

In the tables below, the Sparse Eigenvector algorithm is demonstrated as "SE", the Metropolis Hastings algorithm is represented as "MH", the Heat Bath algorithm is denoted as "HB".

Table 24 Results of the empirical design according to Kendall metric

	Saaty	**SE**	**MH**	**HB**	**SLLS**	**Harker**	**Dematel**	**TLT**	**TLET**
A1	0,21	0,29	0,27	0,30	0,30	0,22	**0,32**	**0,32**	**0,32**
A2	**0,37**	**0,37**	**0,37**	0,30	0,31	**0,37**	**0,37**	**0,37**	**0,37**
A3	0,54	0,52	**0,56**	**0,56**	**0,56**	0,52	0,54	0,54	0,54
A4	0,08	0,09	0,05	0,07	**0,10**	0,09	**0,10**	0,08	0,10
A5	0,10	0,10	0,10	0,10	**0,14**	0,12	0,10	0,10	0,10
A6	**0,54**	0,45	0,45	0,47	0,47	0,43	0,43	0,43	0,43
A7	0,54	0,52	**0,56**	0,54	0,49	0,52	0,50	0,52	0,52
A8	**0,49**	0,16	0,14	0,14	0,16	0,16	0,20	0,22	0,18
A9	0,24	0,28	0,24	**0,31**	0,28	0,28	0,28	0,28	0,28
A10	0,31	0,35	0,33	**0,39**	0,35	0,35	0,37	0,37	0,37
A11	0,28	**0,31**	0,30	0,30	0,30	**0,31**	0,30	0,30	**0,31**
A12	0,16	0,20	0,18	0,20	0,20	0,20	**0,22**	**0,22**	**0,22**
A13	0,16	0,27	0,17	0,23	0,23	**0,29**	0,27	**0,29**	0,27
A14	0,40	0,44	0,44	0,42	**0,45**	0,39	**0,45**	**0,45**	**0,45**
A15	0,10	**0,26**	0,16	0,20	0,24	0,24	**0,26**	**0,26**	**0,26**
Mean	0,302	0,307	0,288	0,302	0,305	0,300	0,314	**0,316**	**0,315**

The algorithm exhibiting the best performance according to Kendall had demonstrated variability across dataset. However, on average, the two algorithms that performed the best were the TLT and TLET algorithms. However, the Metropolis Hastings algorithm displayed the lowest mean performance.

Table 25 Results of the empirical design according to MCIV metric

	SE	MH	HB	SLLS	Harker	Dematel	TLT	TLET
A1	1,020	1,022	1,021	**1,020**	1,020	1,020	1,021	1,020
A2	1,075	1,076	1,074	**1,073**	1,076	1,076	1,076	1,076
A3	1,071	1,080	1,077	**1,070**	1,072	1,071	1,071	1,071
A4	1,022	1,023	1,022	**1,022**	1,023	1,022	1,022	1,022
A5	1,141	1,133	**1,131**	1,132	1,142	1,142	1,142	1,142
A6	1,095	1,097	1,099	**1,094**	1,095	1,095	1,095	1,095
A7	1,079	1,078	1,079	**1,078**	1,080	1,079	1,079	1,079
A8	1,062	1,062	1,060	**1,060**	1,064	1,062	1,063	1,061
A9	1,122	1,128	1,122	**1,121**	1,124	1,124	1,124	1,124
A10	1,072	1,074	1,073	**1,072**	1,072	1,072	1,072	1,072
A11	1,044	1,044	1,044	**1,043**	1,044	1,044	1,044	1,044
A12	1,059	1,060	1,060	**1,058**	1,059	1,059	1,059	1,059
A13	1,011	1,011	1,011	**1,011**	1,011	1,012	1,012	1,011
A14	1,034	1,035	1,035	**1,034**	1,035	1,034	1,035	1,034
A15	1,054	1,064	1,056	**1,053**	1,055	1,058	1,060	1,054
Mean	**1,064**	1,066	1,064	**1,063**	1,065	1,065	1,065	1,064

The algorithm that demonstrates the highest performance based on MCIV did not show variability across datasets. It was the SLLS algorithm as expected due to the objective function of SLLS. On average, the Sparse Eigenvector and the SLLS algorithms were the top two performing algorithms. However, the Metropolis Hastings algorithm exhibited the lowest mean performance.

Table 26 Results of the empirical design according to GCIV-VW metric

	Saaty	SE	MH	HB	SLLS	Harker	Dematel	TLT	TLET
A1	1,661	1,713	**1,646**	1,661	1,702	1,734	1,680	1,676	1,696
A2	1,660	1,627	**1,615**	1,649	1,672	1,618	1,621	1,615	1,622
A3	1,556	1,499	**1,429**	1,433	1,492	1,485	1,478	1,476	1,478
A4	**1,623**	1,711	1,758	1,737	1,741	1,728	1,723	1,725	1,721
A5	1,773	1,765	1,742	**1,740**	1,756	1,767	1,768	1,768	1,768
A6	1,556	**1,398**	1,416	1,432	1,402	1,401	1,401	1,402	1,403
A7	1,227	1,226	**1,223**	1,229	1,237	1,245	1,229	1,228	1,229
A8	**1,340**	1,868	1,826	1,837	1,842	2,146	1,823	1,791	1,859
A9	1,678	1,666	1,646	1,668	1,703	1,652	1,644	**1,644**	1,651
A10	1,526	1,529	1,490	**1,487**	1,524	1,532	1,530	1,530	1,531
A11	1,647	1,643	1,649	1,655	1,668	**1,642**	1,646	1,647	1,644
A12	1,938	1,940	1,928	1,931	**1,923**	1,936	1,932	1,931	1,934
A13	1,825	1,589	1,586	1,589	1,594	**1,583**	1,605	1,605	1,601
A14	1,373	1,374	1,390	1,393	1,373	1,401	**1,371**	1,371	1,374
A15	1,660	1,506	**1,502**	1,504	1,544	1,557	1,524	1,524	1,531
Mean	1,603	1,604	**1,590**	1,596	1,612	1,628	1,598	**1,596**	1,603

The algorithm that shows the highest performance according to GCIV-VW exhibited variability across datasets. The Metropolis Hastings and TLT algorithms are the two algorithms that performed best on average. Conversely, the Harker algorithm demonstrated the lowest mean performance.

After examining the best performances based on each data and mean, the metric performances were compared utilizing statistical tests such as Tukey's HSD and Games Howell in order to analyze statistically significant differences. Consequently, no statistically significant differences were observed among the mean performances of the algorithms. The Tukey's HSD test results for each metric were presented in the appendix chapter (Table 45, 46 and 47).

6. CONCLUDING REMARKS AND FUTURE WORK

The contribution of this research can be summarized in four main aspects. First of all, six algorithms from the incomplete analytic hierarchy process (AHP) literature were introduced and comparatively analyzed in several experimental conditions. Secondly, two novel parametric heuristics, namely TLT and TLET that are suitable for the incomplete AHP framework, were developed. Nine methodologies were generated from these parametric heuristics by assigning different parameters. Three of them by referencing TLT and six of them by referencing TLET were developed. These nine algorithms and the six existing algorithms were also compared in different experimental conditions.

Thirdly, these 15 algorithms' performances were assessed using six metrics that are appropriate for the incomplete AHP framework. Five of these were existing metrics from literature and they were taken into consideration. The remaining metric, namely MCIV, was developed for the incomplete AHP framework in this study. Among these metrics, five of them measure accuracy, and one of them assesses computational time. The relationships between the accuracy metrics were analyzed.

Lastly, two experimental setups consisting of one numerical and one empirical setup were introduced within the incomplete AHP framework. In the numerical experimental setup, 10 different experimental cases were generated based on matrix size, consistency, and missing ratio. In these 10 cases, 15 algorithms were statistically compared according to several metrics, including Kendall's tau, MCIV, GCIV-VW, and central processing unit time. The empirical setup was conducted with the participation of 30 Sabancı University students. In this setup, the 15 incomplete AHP methodologies were statistically compared among themselves based on different metrics including Kendall's tau, MCIV, GCIV-VW.

In addition, the decision maker's attitude towards uncertain pairwise comparisons were also examined by using Kendall' tau and GCIV-VW metrics.

In the numerical experimental design, firstly the three TLT algorithms and six TLET algorithms were statistically compared among themselves and decided the best of the TLT and TLET one. Subsequently, these two selected algorithms and the six existing algorithms were statistically compared. The algorithms generally demonstrated close comparative results in terms of the Kendall, MCIV, and GCIV-VW metrics. However, in some experimental cases, the TLET and the SLSS algorithms exhibited statistical superiority. On the other hand, according to central processing unit time statistical differences among the algorithms were observed in each experimental case. The SLSS algorithm exhibited the worst performance based on central processing unit time due to its convergence time in each case.

In the empirical experimental design, the algorithms exhibited no statistically significant differences. However, there was diversity on the algorithms that perform one of the best two algorithms according to mean in each metric. In contrast to the numerical experimental setup, there is an additional algorithm to understand the decision maker's attitude towards uncertainty. Based on the comparative results of empirical experimental design, it could be preferable for the decision maker to rely on incomplete AHP methodologies than to make predictions about uncertain comparisons.

While the empirical experimental design results showed some alignment with the numerical experimental design results, they also exhibited differences in several metrics performance. For instance, the fact that the TLET algorithm performed among the top two algorithms in terms of the Kendall metric and that the SLSS algorithm was among the top two algorithms in terms of the MCIV metric indicates the consistency of the experimental design outcomes. However, inconsistency arose in the GCIV-VW metric, in which the Metropolis Hastings algorithm performed among the best two algorithms in the empirical experimental design but among the worst two algorithms in the numerical experimental design. This discrepancy might have appeared due to the presence of uncertainties in the real-world situations or factors in the empirical experimental design such as the limited number of participants or their lack of knowledge about the subject of the experiment.

As a future research direction, a sensitivity analysis can be conducted on the parametric heuristics. Different path lengths and various importance parameter values can be analyzed based on the performance metrics and obtained tradeoffs. Another option is to conduct a more comprehensive empirical experiment by involving a larger number of participants with more domain knowledge. An additional option to consider could be exploring relationships between metrics utilizing different principles, aside of correlation such as polynomial or nonlinear relationships.

BIBLIOGRAPHY

Achlioptas, D., Molloy, M., Moore, C., & Van Bussel, F. (2005). Rapid mixing for lattice colourings with fewer colours. *Journal of Statistical Mechanics: Theory and Experiment*, *2005*(10), P10012.

Ahmed, F., & Kilic, K. (2022). A Basic Algorithm for Generating Individualized Numerical Scale (BAGINS). *arXiv preprint arXiv:2211.08740*.

Amiri, M. P. (2010). Project selection for oil-fields development by using the AHP and fuzzy TOPSIS methods. *Expert systems with applications*, *37*(9), 6218-6224.

Bessi, A., Coletto, M., Davidescu, G. A., Scala, A., Caldarelli, G., & Quattrociocchi, W. (2015). Science vs conspiracy: Collective narratives in the age of misinformation. *PloS one*, *10*(2), e0118093.

Bozóki, S., Fülöp, J., & Rónyai, L. (2010). On optimal completion of incomplete pairwise comparison matrices. *Mathematical and computer modelling*, *52*(1-2), 318-333.

Cao, D., Leung, L. C., & Law, J. S. (2008). Modifying inconsistent comparison matrix in analytic hierarchy process: A heuristic approach. *Decision Support Systems*, *44*(4), 944-953.

Carmone Jr, F. J., Kara, A., & Zanakis, S. H. (1997). A Monte Carlo investigation of incomplete pairwise comparison matrices in AHP. *European journal of operational research*, *102*(3), 538-553.

Chamodrakas, I., Batis, D., & Martakos, D. (2010). Supplier selection in electronic marketplaces using satisficing and fuzzy AHP. *Expert systems with applications*, *37*(1), 490-498.

Choo, E. U., Wedley, W. C., & Wijnmalen, D. J. (2016). Mathematical Support for the Geometric Mean when Deriving a Consistent Matrix from a Pairwise Ratio Matrix. *Fundamenta Informaticae*, *144*(3-4), 263-278.

Crawford, G. B. (1987). The geometric mean procedure for estimating the scale of a judgement matrix. *Mathematical Modelling*, *9*(3-5), 327-334.

Crawford, G., & Williams, C. (1985). A note on the analysis of subjective judgment matrices. *Journal of mathematical psychology*, *29*(4), 387-405.

Davis, J. M. (1958). The transitivity of preferences. *Behavioral science*, *3*(1), 26-33.

Dobrow, R. P. (2016). *Introduction to stochastic processes with R*. John Wiley & Sons.

Emrouznejad, A., & Marra, M. (2017). The state of the art development of AHP (1979–2017): A literature review with a social network analysis. *International journal of production research*, *55*(22), 6653-6675.

Fax, J. A., & Murray, R. M. (2004). Information flow and cooperative control of vehicle formations. *IEEE transactions on automatic control*, *49*(9), 1465-1476.

Gomez-Ruiz, J. A., Karanik, M., & Peláez, J. I. (2010). Estimation of missing judgments in AHP pairwise matrices using a neural network-based model. *Applied Mathematics and Computation*, *216*(10), 2959-2975.

Harker, P. T. (1987a). Alternative modes of questioning in the analytic hierarchy process. *Mathematical modelling*, *9*(3-5), 353-360.

Harker, P. T. (1987b). Incomplete pairwise comparisons in the analytic hierarchy process. *Mathematical modelling*, *9*(11), 837-848.

Hayrapetyan, L. R. (2019). RANDOM CONSISTENCY INDICES FOR ANALYTIC HIERARCHY PROCESSES. *International Journal of Business, Marketing, & Decision Science*, *12*(1).

Ishizaka, A., & Labib, A. (2011). Review of the main developments in the analytic hierarchy process. *Expert systems with applications*, *38*(11), 14336-14345.

Kendall, M. G. (1938). A new measure of rank correlation. *Biometrika*, *30*(1/2), 81-93.

Korpela, J., & Lehmusvaara, A. (1999). A customer oriented approach to warehouse network evaluation and design. *International Journal of Production Economics*, *59*(1-3), 135-146.

Kusumawardani, R. P., & Agintiara, M. (2015). Application of fuzzy AHP-TOPSIS method for decision making in human resource manager selection process. *Procedia computer science*, *72*, 638-646.

Lee, S. H. (2010). Using fuzzy AHP to develop intellectual capital evaluation model for assessing their performance contribution in a university. *Expert systems with applications*, *37*(7), 4941-4947.

Lee, S., & Lee, D. K. (2018). What is the proper way to apply the multiple comparison test?. *Korean journal of anesthesiology*, *71*(5), 353-360.

Li, X., Han, Z., Zhang, R., Zhang, Y., & Zhang, L. (2020). Risk assessment of hydrogen generation unit considering dependencies using integrated DEMATEL and TOPSIS approach. *International Journal of Hydrogen Energy*, *45*(53), 29630-29642.

Menci, M., Oliva, G., Papi, M., Setolal, R., & Scala, A. (2018, June). A suite of distributed methodologies to solve the sparse analytic hierarchy process problem. In *2018 European control conference (ECC)* (pp. 1447-1453). IEEE.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., & Teller, E. (1953). Equation of state calculations by fast computing machines. *The journal of chemical physics*, *21*(6), 1087-1092.

Miller, G. A. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological review*, *63*(2), 81.

Olfati-Saber, R., Fax, J. A., & Murray, R. M. (2007). Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, *95*(1), 215-233.

Oliva, G., Setola, R., & Scala, A. (2017). Sparse and distributed analytic hierarchy process. *Automatica*, *85*, 211-220.

Oliva, G., Setola, R., Scala, A., & Dell'Olmo, P. (2019). Sparse analytic hierarchy process: an experimental analysis. *Soft computing*, *23*, 2887-2898.

Ratner, B. (2009). The correlation coefficient: Its values range between+ 1/− 1, or do they?. *Journal of targeting, measurement and analysis for marketing*, *17*(2), 139-142.

Saaty, T. (1972). An eigenvalue allocation model for prioritization and planning. In Working paper, Energy Management and Policy Center, University of Pennsylvania.

Saaty, T. L. (1977). A scaling method for priorities in hierarchical structures. *Journal of mathematical psychology*, *15*(3), 234-281.

Saaty, T. L. (1994). A ratio scale metric and the compatibility of ratio scales: The possibility of arrow's impossibility theorem. *Applied Mathematics Letters*, *7*(6), 51-57.

Saaty, T. L., & Vargas, L. G. (1998). Diagnosis with dependent symptoms: Bayes theorem and the analytic hierarchy process. *Operations research*, *46*(4), 491-502.

Shen, K. Y., & Tzeng, G. H. (2015). Combined soft computing model for value stock selection based on fundamental analysis. Applied Soft Computing, 37, 142–155.

Si, S. L., You, X. Y., Liu, H. C., & Zhang, P. (2018). DEMATEL technique: A systematic review of the state-of-the-art literature on methodologies and applications. *Mathematical Problems in Engineering*, *2018*, 1-33.

Wind, Y., & Saaty, T. L. (1980). Marketing applications of the analytic hierarchy process. *Management science*, *26*(7), 641-658.

Wu, K. J., Liao, C. J., Tseng, M. L., Lim, M. K., Hu, J., & Tan, K. (2017). Toward sustainability: using big data to explore the decisive attributes of supply chain risks and uncertainties. *Journal of Cleaner Production*, *142*, 663-676.

Zhou, X., Hu, Y., Deng, Y., Chan, F. T., & Ishizaka, A. (2018). A DEMATEL-based completion method for incomplete pairwise comparison matrix in AHP. *Annals of Operations Research*, *271*, 1045-1066.

APPENDIX

The tables related to the TLET Comparisons section are as below.

Table 27 Results of the Tukey HSD test for matrix size 5 case based on GCIV-VW

	NEC = 2, imp = 1	NEC = 4, imp = 1	NEC = 0, imp = 2	NEC = 2, imp = 2	NEC = 4, imp = 2
NEC = 0, imp = 1	-0.014 **0.05**	-0.017 **0.004**	0 1.00	-0.01 0.368	-0.012 0.161
NEC = 2, imp = 1		-0.003 0.999	0.014 **0.04**	0.004 0.995	0.002 1.00
NEC = 4, imp = 1			0.017 **0.003**	0.007 0.824	0.005 0.965
NEC = 0, imp = 2				-0.01 0.323	-0.012 0.135
NEC = 2, imp = 2					-0.002 1.00

Table 28 Results of the Tukey HSD Test for med consistency case based on GCIV-VW

	NEC = 2, imp = 1	NEC = 4, imp = 1	NEC = 0, imp = 2	NEC = 2, imp = 2	NEC = 4, imp = 2
NEC = 0, imp = 1	-0.006 0.059	-0.007 **0.005**	0 1.00	-0.005 0.135	-0.006 **0.021**
NEC = 2, imp = 1		-0.001 0.998	0.006 0.056	0.001 1.00	-0.001 1.00
NEC = 4, imp = 1			0.007 **0.005**	0.002 0.983	0.001 1.00
NEC = 0, imp = 2				-0.005 0.13	-0.006 **0.02**
NEC = 2, imp = 2					-0.001 1.00

Table 29 Results of the Tukey HSD test for med consistency case based on MCIV

	NEC = 2, imp = 1	NEC = 4, imp = 1	NEC = 0, imp = 2	NEC = 2, imp = 2	NEC = 4, imp = 2
NEC = 0, imp = 1	-0.005 <**0.001**	-0.007 <**0.001**	0 1.00	-0.005 <**0.001**	-0.006 <**0.001**
NEC = 2, imp = 1		-0.002 0.57	0.005 <**0.001**	0.001 1.00	-0.001 0.973
NEC = 4, imp = 1			0.007 <**0.001**	0.002 0.211	0.001 0.995
NEC = 0, imp = 2				-0.005 <**0.001**	-0.006 <**0.001**
NEC = 2, imp = 2					-0.002 0.753

Table 30 Results of the Tukey HSD test for high consistency case based on MCIV

	NEC = 2, imp = 1	NEC = 4, imp = 1	NEC = 0, imp = 2	NEC = 2, imp = 2	NEC = 4, imp = 2
NEC = 0, imp = 1	-0.01 <**0.001**	-0.013 <**0.001**	0 1.00	-0.008 <**0.001**	-0.01 <**0.001**
NEC = 2, imp = 1		-0.004 0.497	0.01 <**0.001**	0.001 0.996	-0.001 1.00
NEC = 4, imp = 1			0.013 <**0.001**	0.005 0.089	0.003 0.761
NEC = 0, imp = 2				-0.008 <**0.001**	-0.011 <**0.001**
NEC = 2, imp = 2					-0.002 0.952

Table 31 Results of the Tukey HSD test for missing ratio 0.3 case based on MCIV

	NEC = 2, imp = 1	NEC = 4, imp = 1	NEC = 0, imp = 2	NEC = 2, imp = 2	NEC = 4, imp = 2
NEC = 0, imp = 1	-0.005 0.198	-0.007 **0.027**	0 1.00	-0.005 0.336	-0.006 0.079
NEC = 2, imp = 1		-0.002 0.998	0.005 0.19	0.001 1.00	-0.001 1.00
NEC = 4, imp = 1			0.007 **0.026**	0.002 0.987	0.001 1.00
NEC = 0, imp = 2				-0.005 0.325	-0.006 0.076
NEC = 2, imp = 2					-0.001 0.999

Table 32 Results of the Tukey HSD test for missing ratio 0.5 case based on MCIV

	NEC = 2, imp = 1	NEC = 4, imp = 1	NEC = 0, imp = 2	NEC = 2, imp = 2	NEC = 4, imp = 2
NEC = 0, imp = 1	-0.004 0.433	-0.008 **0.002**	0 1.00	-0.003 0.876	-0.004 0.435
NEC = 2, imp = 1		-0.004 0.634	0.005 0.421	0.001 0.999	0 1.00
NEC = 4, imp = 1			0.008 **0.002**	0.005 0.197	0.004 0.631
NEC = 0, imp = 2				-0.003 0.868	-0.005 0.423
NEC = 2, imp = 2					-0.001 0.999

The tables related with All Algorithms Comparisons section are as below.

Table 33 Results of the Tukey HSD test for matrix size 25 case based on Kendall

	Metropolis Hastings	Heat Bath	SLLS	Harker	Dematel	TLT NEC = 0	TLET NEC = 0, imp = 2
Sparse Eigenvector	0.003 0.761	0.001 1.00	-0.003 0.826	0.001 1.00	-0.001 1.00	-0.001 1.00	-0.001 1.00
Metropolis Hastings		-0.002 0.958	-0.006 **0.048**	-0.002 0.943	-0.004 0.524	-0.004 0.524	-0.004 0.459
Heat Bath			-0.004 0.514	-0 1.00	-0.002 0.99	-0.002 0.99	-0.002 0.982
SLLS				0.004 0.56	0.002 0.955	0.002 0.955	0.002 0.973
Harker					-0.002 0.994	-0.002 0.994	-0.002 0.988
Dematel						0 1.00	-0 1.00
TLT-NEC=0							-0 1.00

Table 34 Results of the Tukey HSD test for matrix size 5 case based on MCIV

	Metropolis Hastings	Heat Bath	SLLS	Harker	Dematel	TLT NEC = 0	TLET NEC = 0, imp = 2
Sparse Eigenvector	-0.003 0.071	-0.002 0.751	0 1.00	-0 1.00	-0 1.00	-0.001 0.991	0 1.00
Metropolis Hastings		0.001 0.896	0.003 **0.021**	0.003 0.129	0.003 0.131	0.002 0.439	0.003 0.056
Heat Bath			0.002 0.485	0.001 0.871	0.001 0.873	0.001 0.995	0.002 0.698
SLLS				-0.001 0.999	-0.001 0.998	-0.001 0.92	-0 1.00
Harker					-0 1.00	-0.001 0.999	0 1.00
Dematel						-0.001 0.999	0 1.00
TLT-NEC=0							0.001 0.983

Table 35 The best two algorithms in terms of mean based on Kendall

	N 5	N 15	N 25	MR 0.2	MR 0.3	MR 0.4	MR 0.5	Low Con	Med Con	High Con
Sparse Eigenvector										
Metropolis Hastings										
Heat Bath										
SLLS	X	X	X	X	X	X	X	X	X	X
Harker										
Dematel				X		X				
TLT-NEC = 0	X				X		X		X	
TLET NEC = 0, imp = 2		X	X					X		X

Table 36 The worst two algorithms in terms of mean based on MCIV

	N 5	N 15	N 25	MR 0.2	MR 0.3	MR 0.4	MR 0.5	Low Con	Med Con	High Con
Sparse Eigenvector										
Metropolis Hastings	X	X	X	X	X	X	X	X	X	X
Heat Bath	X	X	X	X	X	X	X	X	X	X
SLLS										
Harker										
Dematel										
TLT-NEC = 0										
TLET NEC = 0, imp = 2										

Table 37 The worst two algorithms in terms of mean based on Kendall

	N 5	N 15	N 25	MR 0.2	MR 0.3	MR 0.4	MR 0.5	Low Con	Med Con	High Con
Sparse Eigenvector										
Metropolis Hastings	X	X	X	X	X	X	X		X	X
Heat Bath		X	X	X	X	X		X	X	
SLLS										
Harker	X						X			X
Dematel										
TLT-NEC = 0								X		
TLET NEC = 0, imp = 2										

Table 38 Results of the Games Howell test for matrix size 15 case based on central processing unit time

	Metropolis Hastings	Heat Bath	SLLS	Harker	Dematel	TLT NEC = 0	TLET NEC = 0, imp = 2
Sparse Eigenvector	-0.019 **<0.001**	-0.046 **<0.001**	-0.064 **<0.001**	-0 0.997	-0 1.00	-0 0.058	-0.003 **<0.001**
Metropolis Hastings		-0.026 **0.007**	-0.044 **<0.001**	0.019 **<0.001**	0.019 **<0.001**	0.019 **<0.001**	0.016 **<0.001**
Heat Bath			-0.018 0.545	0.046 **<0.001**	0.046 **<0.001**	0.045 **<0.001**	0.042 **<0.001**
SLLS				0.064 **<0.001**	0.064 **<0.001**	0.064 **<0.001**	0.061 **<0.001**
Harker					0 1.00	-0 0.296	-0.003 **<0.001**
Dematel						-0 0.144	-0.003 **<0.001**
TLT-NEC=0							-0.003 **<0.001**

Table 39 Results of the Games Howell test for low consistency case based on central processing unit time

	Metropolis Hastings	Heat Bath	SLLS	Harker	Dematel	TLT NEC = 0	TLET NEC = 0, imp = 2
Sparse Eigenvector	-0.025 **<0.001**	-0.042 **<0.001**	-0.098 **<0.001**	-0 0.703	-0 0.703	-0.001 **<0.001**	-0.011 **<0.001**
Metropolis Hastings		-0.017 0.08	-0.074 **<0.001**	0.025 **<0.001**	0.025 **<0.001**	0.024 **<0.001**	0.013 **0.001**
Heat Bath			-0.057 **0.017**	0.042 **<0.001**	0.042 **<0.001**	0.041 **<0.001**	0.03 **<0.001**
SLLS				0.098 **<0.001**	0.098 **<0.001**	0.098 **<0.001**	0.087 **<0.001**
Harker					0 1.00	-0.001 **0.002**	-0.011 **<0.001**
Dematel						-0.001 **0.002**	-0.011 **<0.001**
TLT-NEC=0							-0.011 **<0.001**

Table 40 Results of the Games Howell test for med consistency case based on central processing unit time

	Metropolis Hastings	Heat Bath	SLLS	Harker	Dematel	TLT NEC = 0	TLET NEC = 0, imp = 2
Sparse Eigenvector	-0.019 <**0.001**	-0.045 <**0.001**	-0.094 <**0.001**	-0 1.00	0 1.00	-0 **0.024**	-0.011 <**0.001**
Metropolis Hastings		-0.026 <**0.001**	-0.075 <**0.001**	0.019 <**0.001**	0.019 <**0.001**	0.019 <**0.001**	0.008 <**0.001**
Heat Bath			-0.049 **0.015**	0.045 <**0.001**	0.045 <**0.001**	0.044 <**0.001**	0.033 <**0.001**
SLLS				0.094 <**0.001**	0.094 <**0.001**	0.093 <**0.001**	0.082 <**0.001**
Harker					0 1.00	-0 **0.033**	-0.011 <**0.001**
Dematel						-0.0005 **0.024**	-0.011 <**0.001**
TLT-NEC=0							-0.011 <**0.001**

Table 41 Results of the Games Howell test for missing ratio 0.2 case based on central processing unit time

	Metropolis Hastings	Heat Bath	SLLS	Harker	Dematel	TLT NEC = 0	TLET NEC = 0, imp = 2
Sparse Eigenvector	-0.02 <**0.001**	-0.037 <**0.001**	-0.079 <**0.001**	-0 1.00	-0 0.805	-0 0.442	-0.012 <**0.001**
Metropolis Hastings		-0.018 0.075	-0.06 **0.005**	0.02 <**0.001**	0.019 <**0.001**	0.019 <**0.001**	0.008 0.07
Heat Bath			-0.042 0.192	0.037 <**0.001**	0.037 <**0.001**	0.037 <**0.001**	0.025 <**0.001**
SLLS				0.079 <**0.001**	0.079 <**0.001**	0.079 <**0.001**	0.067 <**0.001**
Harker					-0 0.921	-0 0.625	-0.012 <**0.001**
Dematel						-0 0.999	-0.012 <**0.001**
TLT-NEC=0							-0.012 <**0.001**

Table 42 Results of the Games Howell test for missing ratio 0.3 case based on central processing unit time

	Metropolis Hastings	Heat Bath	SLLS	Harker	Dematel	TLT NEC = 0	TLET NEC = 0, imp = 2
Sparse Eigenvector	-0.017 **<0.001**	-0.049 **<0.001**	-0.084 **<0.001**	-0 0.999	-0 1.00	-0.001 **<0.001**	-0.012 **<0.001**
Metropolis Hastings		-0.031 **0.004**	-0.067 **<0.001**	0.017 **<0.001**	0.017 **<0.001**	0.017 **<0.001**	0.006 **0.014**
Heat Bath			-0.036 0.38	0.049 **<0.001**	0.049 **<0.001**	0.048 **<0.001**	0.037 **<0.001**
SLLS				0.084 **<0.001**	0.084 **<0.001**	0.084 **<0.001**	0.073 **<0.001**
Harker					0 1.00	-0.001 **0.007**	-0.012 **<0.001**
Dematel						-0.001 **0.004**	-0.012 **<0.001**
TLT-NEC=0							-0.011 **<0.001**

Table 43 Results of the Games Howell test for missing ratio 0.4 case based on central processing unit time

	Metropolis Hastings	Heat Bath	SLLS	Harker	Dematel	TLT NEC = 0	TLET NEC = 0, imp = 2
Sparse Eigenvector	-0.023 **<0.001**	-0.044 **<0.001**	-0.091 **<0.001**	-0 0.205	0 1.00	-0.001 **0.005**	-0.011 **<0.001**
Metropolis Hastings		-0.021 0.135	-0.068 **0.004**	0.022 **<0.001**	0.023 **<0.001**	0.022 **<0.001**	0.011 **0.011**
Heat Bath			-0.047 0.218	0.044 **<0.001**	0.044 **<0.001**	0.044 **<0.001**	0.033 **<0.001**
SLLS				0.09 **<0.001**	0.091 **<0.001**	0.09 **<0.001**	0.079 **<0.001**
Harker					0 0.075	-0 0.905	-0.011 **<0.001**
Dematel						-0.001 **0.001**	-0.012 **<0.001**
TLT-NEC=0							-0.011 **<0.001**

Table 44 Results of the Games Howell test for missing ratio 0.5 case based on central processing unit time

	Metropolis Hastings	Heat Bath	SLLS	Harker	Dematel	TLT NEC = 0	TLET NEC = 0, imp = 2
Sparse Eigenvector	-0.028 **<0.001**	-0.046 **<0.001**	-0.097 **<0.001**	-0 1.00	0 1.00	-0.001 **<0.001**	-0.01 **<0.001**
Metropolis Hastings		-0.018 0.102	-0.068 **<0.001**	0.028 **<0.001**	0.028 **<0.001**	0.027 **<0.001**	0.018 **0.007**
Heat Bath			-0.051 **0.012**	0.046 **<0.001**	0.046 **<0.001**	0.045 **<0.001**	0.036 **<0.001**
SLLS				0.097 **<0.001**	0.097 **<0.001**	0.096 **<0.001**	0.086 **<0.001**
Harker					0 0.999	-0.001 **<0.001**	-0.01 **<0.001**
Dematel						-0.001 **<0.001**	-0.01 **<0.001**
TLT-NEC=0							-0.009 **<0.001**

The tables related to the Empirical Experimental Design Results section are as below.

Table 45 Results of the Tukey HSD test based on Kendall in empirical design

	Sparse Eigenvector	Metropolis Hastings	Heat Bath	SLSS	Harker	Dematel	TLT NEC=0	TLET NEC=0, imp=2
Saaty	-0.006 1.00	0.013 1.00	-0.001 1.00	-0.004 1.00	0.002 1.00	-0.013 1.00	-0.015 1.00	-0.013 1.00
Sparse Eigenvector		0.019 1.00	0.005 1.00	0.002 1.00	0.008 1.00	-0.007 1.00	-0.009 1.00	-0.007 1.00
Metropolis Hastings			-0.014 1.00	-0.017 1.00	-0.011 1.00	-0.026 1.00	-0.029 1.00	-0.027 1.00
Heat Bath				-0.003 1.00	0.003 1.00	-0.012 1.00	-0.015 1.00	-0.013 1.00
SLSS					0.006 1.00	-0.009 1.00	-0.011 1.00	-0.009 1.00
Harker						-0.015 1.00	-0.017 1.00	-0.015 1.00
Dematel							-0.003 1.00	-0.001 1.00
TLT-NEC=0								0.002 1.00

Table 46 Results of the Tukey HSD test based on MCIV in empirical design

	Metropolis Hastings	Heat Bath	SLLS	Harker	Dematel	TLT NEC = 0	TLET NEC = 0, imp = 2
Sparse Eigenvector	-0.002 1.00	-0 1.00	0.001 1.00	-0.001 1.00	-0.001 1.00	-0.001 1.00	-0 1.00
Metropolis Hastings		0.002 1.00	0.003 1.00	0.001 1.00	0.001 1.00	0.001 1.00	0.002 1.00
Heat Bath			0.002 1.00	-0.001 1.00	-0 1.00	-0.001 1.00	0 1.00
SLLS				-0.002 1.00	-0.002 1.00	-0.002 1.00	-0.002 1.00
Harker					0 1.00	-0 1.00	0.001 1.00
Dematel						-0 1.00	0 1.00
TLT-NEC=0							0.001 1.00

Table 47 Results of the Tukey HSD test based on GCIV-VW in empirical design

	Sparse Eigenvector	Metropolis Hastings	Heat Bath	SLSS	Harker	Dematel	TLT NEC=0	TLET NEC=0, imp=2
Saaty	-0.001 1.00	0.013 1.00	0.007 1.00	-0.009 1.00	-0.026 1.00	0.004 1.00	0.007 1.00	0 1.00
Sparse Eigenvector		0.014 1.00	0.007 1.00	-0.008 1.00	-0.025 1.00	0.005 1.00	0.008 1.00	0.001 1.00
Metropolis Hastings			-0.007 1.00	-0.022 1.00	-0.039 1.00	-0.009 1.00	-0.006 1.00	-0.013 1.00
Heat Bath				-0.015 1.00	-0.032 1.00	-0.002 1.00	0.001 1.00	-0.007 1.00
SLSS					-0.017 1.00	0.013 1.00	0.016 1.00	0.009 1.00
Harker						0.030 1.00	0.033 1.00	0.026 1.00
Dematel							0.003 1.00	-0.004 1.00
TLT-NEC=0								-0.007 1.00