國立交通大學

資訊管理研究所

博 士 論 文

以模糊邏輯建構的

品質導向網際網路服務選擇模型

A QoS-Based Web Service Selection Model

Using Fuzzy Logic

研 究 生：林謂立

指導教授：羅濟群 教授

共同指導教授：趙國銘 教授

中 華 民 國 九 十 八 年 六 月

以模糊邏輯建構的
品質導向網際網路服務選擇模型

A QoS-Based Web Service Selection Model
Using Fuzzy Logic

研 究 生 ：林謂立 Student : Wei-Li Lin

指導教授 ： 羅濟群 Advisor : Chi-Chun Lo

共同指導教授 ： 趙國銘 Co-Advisor : Kuo-Ming Chao

國 立 交 通 大 學
資 訊 管 理 研 究 所
博 士 論 文

A Dissertation
Submitted to Institute of Information Management
College of Management
National Chiao Tung University
in partial Fulfillment of Requirements
for the degree of
Doctor of Philosophy
in
Information Management
May 2009
Hsinchu, Taiwan, Republic of China

中 華 民 國 九 十 八 年 六 月

以模糊邏輯建構的
品質導向網際網路服務選擇模型

學生：林謂立

指導教授：羅濟群
共同指導教授：趙國銘

國立交通大學資訊管理研究所博士班

摘 要

在本研究中，我們進行了兩階段以網際網路服務品質為導向的網際網路服務選擇模型研究 - QCMA 模型(QCMA: QoS Consensus Moderation Approach) 與 FMG-QCMA 模型(Fuzzy Multi- Groups-Based QCMA)。第一階段的 QCMA 模型著重以辨識網際網路參與者對網際網路服務品質感知的相似度，進而確認這些參與者是否具高相似度，並根據已確認之高相似度參與者對網際網路服務品質因素偏好優先順序，決定這個高相似性群體對於網際網路服務選擇之決策模型。第二階段的 FMG-QCMA 模型著重於思考不同之消費者其差異過大的性格背景偏好差異，進而研究結合多維度的網際網路服務品質因素結構的分群演算法，建立更有效率的多群組架構網際網路選擇的決策模型。同時，在分群結構之中，因相似度些微低於分群比對的相似合格度，就被裁定網際網路服務品質意見為不相似，本研究也提出了模糊邊界的概念，將近似合格邊界的網際網路品質意見，納入為該群組之模糊相似意見，進而更明確的掌握分群有效性，避免分群失真現象。

本研究之模型適合任一種網際網路服務應用之目標客群分析，如線上旅行社、網際網路商城、網路拍賣會等網際網路應用服務。

關鍵字：品質服務，網際網路服務，模糊邏輯，服務決策

A QoS-Based Web Service Selection

Using Fuzzy Logic

Student : Wei-Li Lin

Advisor : Chi-Chun Lo
Co-Advisor : Kuo-Ming Chao

Institute of Information Management

National Chiao Tung University

ABSTRACT

In this research, two stages of modeling for QoS-aware selection of web service were established – QCMA (QoS Consensus Moderation Approach) and FMG-QCMA (Fuzzy Multi-Groups-Based QCMA). QCMA was proposed as the first stage in order to indentifying a group of participants by their high similarity and obtaining the group preference over all QoS attributes. FMG-QCMA was proposed as second stage in order to thinking over the distinct background and preference over QoS attributes among all web service participants. For this purpose a more efficient multi-attributes-based multi-groups clustering approach was studied for developing multi-groups-based QoS-aware selection model of web service. Also, the concept of fuzzy boundary, which is used for preventing possible omission of some opinions that should be treated as "similar" to group centre but cannot beyond the threshold distance defined in clustering criterion, was thought over.

The models in the research can be applied to "target customers analysis" on any web service application such as e-tourist agency, e-mall or e-auction.

Keywords: QoS, Web Service, Fuzzy Logic, Service Selection

誌 謝

能夠修得博士學位的桂冠，尤其能在交通大學這樣一流的學府實現這個夢想，這並不是年少時期的我所能夠想像的事。所以當通過博士論文最終口試的那一刻，跟五位口試委員老師一一握手致謝時，我心裡的快樂與感動是言語很難形容的。

能夠完成交大博士這個夢想，首先要感謝的是我的指導教授羅濟群博士，因為他是實現這個夢想最重要的推手。從 2001 年我進入交大的管理學院碩士專班進修碩士學位，到考取資管所博士班進修博士學位，這八年來我都在羅老師的鼓勵支持與治學嚴謹思維帶領下，一點一滴地建立起一個博士應有的獨立、完整、與縝密的研究態度與基礎。而另一位要感謝的重要導師，也就是我的共同指導教授，英國 Coventry University 的趙國銘博士。過去五年在趙老師與羅老師的指導下，我們與英國 Oxford Brookes University 的 Dr. Muhammad Younas、Coventry University 的 Dr. Nick Godwin 合作，分別發表了兩篇 SCI / A 級期刊論文與兩篇重量級的學術會議論文，為我的博士學位奠定了無比重要的基礎。而這五年來多次與趙老師越洋挑燈夜戰式的討論，面對趙老師對數學理論一絲不苟的學術要求，以及跟趙老師一起在維也納與福岡面對最嚴格的學術檢視，更建立了我與趙老師無比深刻的革命情感。

在過去這六年的博士學位進修中，感謝資管所所有的老師們的細心教導，由於受教於陳安斌老師、曾國雄老師、劉敦仁老師、游伯龍老師、黎漢林老師、楊千老師、陳瑞順老師、黃景彰老師、林妙聰老師、蔡銘箴老師，為我的博士學位進修奠定了完整的視野。我還要感謝已畢業的王平學長跟我的同班同學佳楨，他們在我期刊論文與博士論文撰稿時，提供了許多重要的經驗與實質的協助，是我完成論文最後階段非常重要的支持力量。同時，我也要感謝通訊網路研究室的博士班學弟鼎元與所辦公室的淑惠，從博士論文計劃書與口試這幾個月，替我分攤了許多博士論文審查與畢業典禮許多重要的準備工作。我還要特別感謝我的博士班同班兄弟姐妹們，彥曲、美奐、佳楨、端芳、振生、文

智、浩鈞、先志、嘉輝、耿杰，謝謝你們對我這個班代的支持與鼓勵，在博士生資格考試期間，跟彥曲、美奐、佳禎、文智、浩鈞、振生、耿杰、嘉輝狂搞讀書會，除了晚上不睡覺在 MSN 上討論人工智慧習慣領域，時而像蟑螂一樣四處吃香喝辣同時解數學題，還有跟彥曲一起寫會議論文投中 ICEB2004，以及全班班費湊一湊，沒事就找奇怪的餐廳吃美食 ……，因為有你們，這一切一切編織了我在交大博士進修最美好的記憶，而且這個記憶顯然會因為我們堅定的友誼，將跨越博士學位的獲得，一直美好的走下去。

我還要深切的感謝這次擔任我博士論文最終口試的政大楊建民老師、中正大學黃興進老師、本所所長劉敦仁老師、還有所上的李永銘老師，在這次博士論文計劃書口試與博士論文最終口試中，給所非常多洞見深遠的寶貴意見，而這些指導無疑的將會成為我未來研究方向非常重要的參考。

最後，我要感謝我的父母、岳父母、與家中長輩與兄弟姐妹的支持，尤其是內人肇芳，除了在我進修碩士、博士學位這八年來，替我照顧好一雙兒女，讓我無後顧之憂，而且她在電信資訊產業的專業看法與分析能力，也幫助我解決不少研究上的難題，這次博士論文簡報的製作，當中就有許多來自她的建言，所以能有她這位愛人同志的支持，是我這次博士學位進修最大的動力。

完成博士學位的進修，其實是另一個研究生涯的開始，尤其我確切的明白自己仍有太多還要學習與成長的地方。要感謝的人其實遠遠不止於寫在誌謝裡面的師長友人，我願意永遠懷抱這感謝的心情，貢獻所學，也將所學傳承下去，企盼能在未來的大環境裡，讓這個博士桂冠發光發熱，即使微不足道，但卻是全心全意的，給予。

Content

Table

Figure

Symbols

1. $wsa_{a_i}^{k}$: The fuzzy QoS opinions represented by each web service participant (consumer *k*'s fuzzy QoS opinion) regarding his/her subjective fuzzy preference on each specific QoS attribute (a_i) in format of a positive trapezoidal fuzzy number, which can be represented as:

$$wsa_{a_i}^{k} = ((x_1)_{a_i}^{k}, (x_2)_{a_i}^{k}, (x_3)_{a_i}^{k}, (x_4)_{a_i}^{k}),\ 0 \leqq (x_1)_{a_i}^{k} \leqq (x_2)_{a_i}^{k} \leqq (x_3)_{a_i}^{k} \leqq (x_4)_{a_i}^{k} \leqq 10 \quad (1)$$

2. WSA_{a_i} : The collection of all $wsa_{a_i}^{k}$ for QoS attribute α_i, which can be represented as:

$$WSA_{a_i} = \{ wsa_{a_i}^{k} \mid k \in K, \alpha_i \in S_Q\}, \qquad S_Q = \{a_1, a_2, a_3, \ldots, a_{13}\} \quad (2)$$

3. $wsa_{S_Q}^{k}$: Multi-attributes-based fuzzy QoS opinion for consumer *k* with QoS attributes in S_Q.

$$wsa_{S_Q}^{k} = (wsa_{a_1}^{k}, wsa_{a_2}^{k}, \ldots, wsa_{a_{13}}^{k}) \quad (3)$$

4. WSA_{S_Q} : is the set of all collected $wsa_{S_Q}^{k}$ which can be represented as follow.

$$WSA_{S_Q} = \left\{ wsa_{S_Q}^{k} \middle| k \in K, a_i \in S_Q \right\} \quad (4)$$

5. G_p: is p^{th} sub-group clustered by algorithm *Fuzzy_Clustering* and *Clustering_Verification*. In G_p there are two subset defined: G_p*.Abs_Sim* is defined as the set with the fuzzy QoS opinions which have full membership with sub-group centre in similarity; G_p*.Fuz_Sim* is defined as the set with the fuzzy QoS opinions which have partial membership with sub-group centre in similarity.

6. $wsa_{a_i}^{G_p}$: is the sub-group centre of G_p on dimension of QoS attribute a_i.

7. $wsa_{S_Q}^{G_p}$: is the sub-group centre of G_p for all QoS attributes in S_Q.

8. $WSA_{S_Q}^{G_p}$: is the set of all collected $wsa_{S_Q}^{k}$ in G_p which can be represented as follow.

$$WSA_{S_Q}^{G_p} = \left\{ wsa_{S_Q}^{k} \middle| k \in K, wsa_{S_Q}^{k} \in G_p \right\} \tag{5}$$

9. n_t : The number of all $wsa_{S_Q}^{k}$ in WSA_{S_Q} collected on time t.

10. $n_t^{G_p}$: The number of all $wsa_{S_Q}^{k_{G_p}}$ in $WSA_{S_Q}^{G_p}$, that is:

$$n_t = \sum_{i=1}^{m} n_t^{G_p} \qquad \text{for } (G_1, G_2, \ldots, G_m),\ 1 \leqq p \leqq m. \tag{6}$$

11. $o_{a_i}^{k}$: Relative position of preference order for $wsa_{a_i}^{k}$ in $wsa_{S_Q}^{k}$.

12. $Sim_{a_i}^{jk}$: Single attribute-based similarity between $wsa_{a_i}^{j}$ and $wsa_{a_i}^{k}$ which can be represented as follow.

$$Sim_{a_i}^{jk} = \frac{\min\left\{\left((x_1)_{a_i}^{j} + \int \tilde{\mu}(wsa_{a_i}^{j})dx\right), \left((x_1)_{a_i}^{k} + \int \tilde{\mu}(wsa_{a_i}^{k})dx\right)\right\}}{\max\left\{\left((x_1)_{a_i}^{j} + \int \tilde{\mu}(wsa_{a_i}^{j})dx\right), \left((x_1)_{a_i}^{k} + \int \tilde{\mu}(wsa_{a_i}^{k})dx\right)\right\}} \tag{7}$$

13. $Sim_{S_Q}^{jk}$: Multi-attributes-based Similarity between $wsa_{S_Q}^{k}$ and $wsa_{S_Q}^{j}$ which can be represented as follow.

$$Sim_{S_Q}^{jk} = (so_{a_1}^{jk} \times Sim_{a_1}^{jk}, so_{a_2}^{jk} \times Sim_{a_2}^{jk}, \ldots, so_{a_{13}}^{jk} \times Sim_{a_{13}}^{jk}) \tag{8}$$

where $so_{a_i}^{jk}$ indicates weight for each $Sim_{a_i}^{jk}$ which can be obtained by equation (9):

$$so_{a_i}^{jk} = \frac{q - \left|o_{a_i}^{j} - o_{a_i}^{k}\right|}{q} \tag{9}$$

q indicates the number of QoS attributes. By definition in W3C [40], $q = 13$.

14. $\widetilde{d}_{S_Q}$: is a pair of similarity thresholds given by an expert to emphasis the more extreme similarities given by the components of (8) in the selection and rejection of consumers for clusters.

$$\widetilde{d}_{S_Q} = (0,\ d^l_{S_Q}, d^u_{S_Q},\ 1) = (d^l_{S_Q}, d^u_{S_Q}) \qquad 0 \leqq d^l_{S_Q} < d^u_{S_Q} \leqq 1 \tag{10}$$

15. $\widetilde{d}'_{S_Q}$: is also a similarity threshold in form of a fuzzy trapezoidal number, which could be moderated by service consumers' later feedback for adjustment.

$$\widetilde{d}'_{S_Q} = (0,\ d'^l_{S_Q},\ d'^u_{S_Q},\ 1) = (d'^l_{S_Q},\ d'^u_{S_Q}), \qquad 0 \leqq d'^l_{S_Q} < d'^u_{S_Q} \leqq 1 \tag{11}$$

16. $f_{c\_S_Q}$: is the threshold for evaluating *sim_result* in algorithm *SimVerifier*.

$$f_{c\_S_Q} = (f^l_{c\_S_Q},\ f^u_{c\_S_Q}), \qquad 0 \leqq f^l_{c\_S_Q} < f^u_{c\_S_Q} \leqq 1 \tag{12}$$

Chapter 1. Introduction - QoS-aware Selection of Web Services

1.1 Research Background and Motivation

In the booming internet world, Web services have become a promising technology for e-trading in recent years. A number of Internet-based software systems such as hotel booking application, e-auction, e-mall, e-air ticket booking, etc., were deployed increasingly (some of them could be integrated) over the world [13][28][30]. For instance, a hotel booking application can be exposed as a web service and integrated with other applications such as flight booking or car-rental in order to provide an integrated environment for web service consumers. However, there exist a large number of similar web services provided by different web service providers, such as e-tourist agencies, which compose similar web services on aspect of functional characteristics such as web services information matchmaking (about pricing, facilities, breakfast, etc.) by consumers' inquiries. Therefore, consumers' behavior regarding how they select a web service was addressed in many researches [3][4][39][40].

However, Web services could be not selected by the consumers if the way how the web services would be provided is far away from what the consumers expect, even though the answers replied by the web services system are what the consumers want. For those consumers who care about quality of service (QoS) when using web service such as efficiency, interoperability and system reliability when they using web services could feel uncomfortable, even be roiled, if they have to make a choice among thousands web services with much longer time than expectation. For instance, a consumer Mr. White who prefers high performance and interoperability in web service issues an inquiry for recommended cheap hotels from e-tourist web service. However, the e-tourist returns more than 1,500 sets of answer with around 5 seconds for this inquiry, such as the situation in Figure 1. Assume Mr. White just wants to catch less than 10 sets of answer in 0.05 second. It could make Mr. White feel impatient on the returned 1,500 sets of answer with around 5 seconds so that he would reject to use the web service.

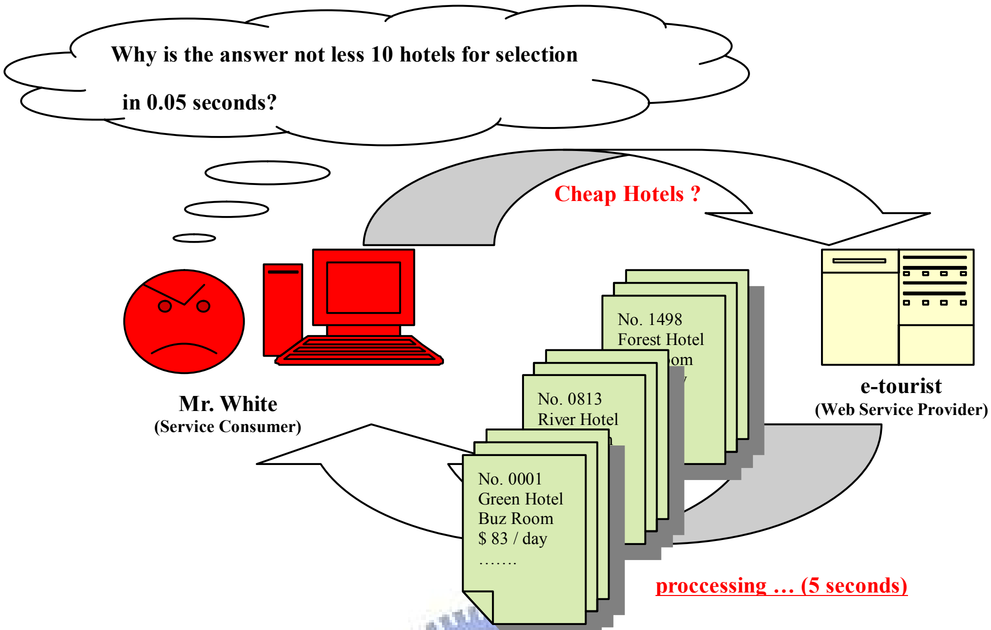

Figure 1: Inquiry for Recommended Web Service far away from Expectation

The example in Figure 1 hints that consumers will no longer select a web service if the web service systems just focus on the provisioning of "right answers" according to consumers' inquiry through information matchmaking. Definitely, no service providers would like to anger consumers by inappropriate interaction. Therefore, service consumers' disposition on QoS of web service should be aware of.

For catching a group of "target customers", service consumers' preference over QoS attributes in web service becomes a key topic. However, the service consumers' preference over QoS should be evaluated on the basis of similarity in consumers' disposition about QoS. For instance, they may have distinct views of the service reliability — wherein a consumer considers that a service is reliable if its success rate is higher than 99%, while a provider may consider its service as reliable if its success rate is just higher than 90%. The conflicts, which could be very hard to settle, should be resolved via multi-groups-based QoS preference analysis over framework of multi-QoS attributes and multi-groups basis, if it could happen for different groups of consumers on different QoS attributes simultaneously. Definitely, the consensus on the QoS characteristics in the selection of web services for a specific group of consumers should be ensured.

It is why we do the research to devising techniques to publish subjective QoS values to assist service consumers in selecting services according to the desired level of QoS. If the preference over QoS attributes for a group of consumers could be caught in advance by web services system, then the “target customers” for web service providers can be identified. Regarding QoS, in this research we follow the QoS standard terms announced by W3C in 25, November, 2003 [40].

1.2 Two Stages for Research Objective: QCMA and FMG-QCMA

For establishing a more effective web service selection model and resolving the issues regarding QoS-aware selection of web service through literatures research, there were two stages of modeling for this research objective – QCMA and FMG-QCMA.

QCMA (QoS Consensus Moderation Approach) is proposed as a consensus-based model for QoS moderation approach which was derived from MFDM (Moderated Fuzzy Discovery Method) [22] with SAM (Similarity Aggregation Method) and RMGDP (Resolution Method for Group Decision Problems) [9][10][11][18][19]. Based on the fuzzy matchmaking approach carried out by MFDM for consensus on discovered web service information between web service consumers and web service providers on functional aspects of fuzzy inquiries in applications (such as “very cheap”, “comfortable”, “delicious”, the factors in terms of pricing, facilities and food), QCMA enhances the architecture / mechanism of MFDM for a group of web service consumers to reach consensus on QoS for web service which were defined by W3C [40], which can be depicted as following figure.

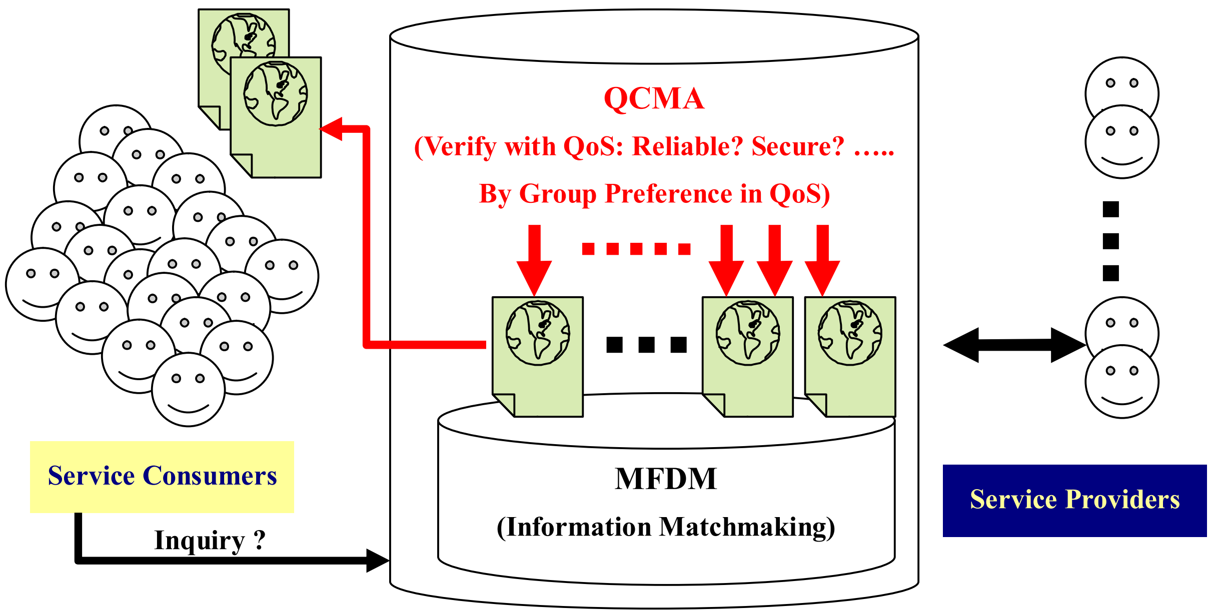

Figure 2: The Functional Enhancement from MFDM to QCMA

Besides, QCMA also moderates their preferences and expectations in order to have coherent definitions of QoS characteristics using fuzzy terms. That is, web service consumers can express their QoS requirements using fuzzy terms such as 'very reliable' and 'less efficient'. Based on the group consensus on QoS characteristics, the group preferences order over all given QoS attributes can be obtained as the criterion how the group of consumers would select a web service.

FMG-QCMA (Fuzzy Multi-Groups-based QCMA) , an evolved extension of QCMA framework by incorporating fuzzy clustering mechanism, attempts to provide a more effective architecture / mechanism for fuzzy multi-groups-based web service selection. All the incoming fuzzy multi-attributes-based QoS opinions will be fuzzily clustered into fuzzy opinion sub-groups (fuzzy sets with objects in multi-attributes fuzzy trapezoidal number format) according to given fuzzy clustering criteria. With the clustered multi-groups structure, all clustered multi-attributes-based QoS opinions will be further analyzed by FMGSAM (Fuzzy Multi-Groups-based SAM, derived from SAM but for multi-groups structure) and multi-groups-based RMGDP. FMGSAM is also the main mechanism in FMG-QCMA which allows the system to be more effective, efficient and flexible in QoS-aware web service selection than QCMA.

The web service model through FMG-QCMA can be depicted as following figure.

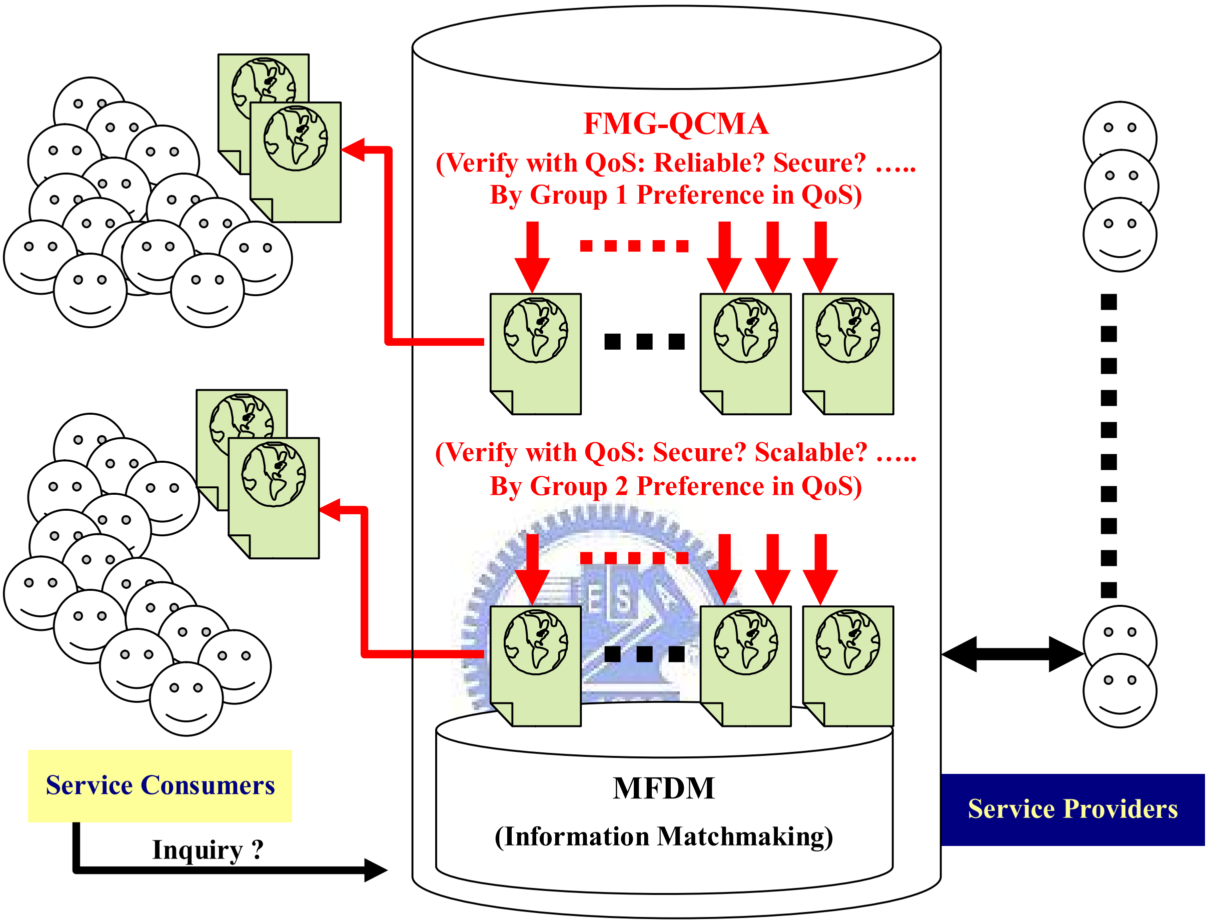

Figure 3: The Multi-Groups-based Web Service Selection via FMG-QCMA

For carrying out the novel model of QoS-aware selection of web service by the motivation mentioned in previous section, the main research objective by the two stages of modeling can be summarized as follow:

1. To provide an architecture / mechanism for group(s) of web service consumers to reach consensus on the definitions of QoS characteristics. Regarding the QoS terms, the QoS characteristics defined by W3C [40] will be adopted.

2. The proposed approach moderates their preferences and expectations in order to have

coherent definitions of QoS characteristics using fuzzy terms. That is, service consumers can express their QoS requirements using fuzzy terms such as 'very reliable' and 'high efficiency'.

3. The group consensus will be built upon opinion similarity and QoS preference order. For this purpose, web service consumers will express his / her subjective opinion on each QoS attribute with fuzzy scale of perception (will be transformed as trapezoidal fuzzy number) and his / her preference in order among all given QoS attributes.

4. The group preference order among QoS attributes will be taken as the preference to select web service. Therefore, the group preference order must be obtained from the group which comprises consumers who have similar opinions in QoS attributes.

5. For dealing multi-QoS attributes opinions from large number of consumers in internet world, the whole model should be executed in excellent efficiency with lower operational complexity.

Through carrying out the research objectives above, QCMA and FMG-QCMA can be established with organized system framework and efficient system operation. The group preference how consumers select web service can be obtained accordingly. Therefore, through two stages from QCMA to FMG-QCMA, QoS was proven as the key factors to drive web service selection. With fuzzy analysis in terms of QoS opinions similarity and QoS preference, any kind of consumer perception distribution can be effectively / efficiently modelled so that most kind of web service scenario can be performed in terms of QoS preference for specific customers group which has high degree of consensus in QoS similarity.

1.3 Structure of the Dessertation

The structure of the dissertation for this thesis can be introduced as follow:

Chapter 2 describes the literatures review on traditional research regarding QoS-aware selection of a web service and multi-attributes-based opinions clustering approaches.

Chapter 3 describes how QCMA was developed as a unique group-based QoS-aware model of web service selection with theories of SAM and RMGDP, reports on experimental results with a case study of a hotel booking web service selection for QCMA, and conclude the pros and cons about QCMA.

Chapter 4 presents how the multi-groups framework, FMG-QCMA, was evolved from QCMA. Similar to chapter 3 for QCMA, a case study of a hotel booking web service selection is introduced. Thereafter, the improvement given by FMG-QCMA than QCMA for QoS-aware selection of web service is concluded.

Chapter 5 makes the conclusion for the series study of QoS-aware selection of web service. After all, some future works will be suggested.

Chapter 2. Literatures Review

For reaching the objective of QoS-aware selection of web services, whether the involved consumers could have similar perception or could be quite diversified in their background, a number of researches regarding web service, existed QoS-aware selection of web services and multi-attributes-based opinions clustering were surveyed in this study. These surveyed researches will be introduced respectively as follow.

2.1 Web Service

2.1.1 Web Service Definition

In general, a Web Service is seen as an application accessible to the other applications over the Web. However, the simple definition could be not definite enough so that the term *Web service* is used very often nowadays but not always with the same meaning. Therefore, in this research the definition of Web Service proposed by W3C was followed and can be stated as follow [1]:

> ***A software application identified by a URI, whose interfaces and bindings are capable of being defined, described, and discovered as XML artifacts. A Web service supports directly interactions with other software agents using XML-based messages exchanged via Internet-based protocols.***

For implementing Web Service described above, a lot of open standards have had been used which including XML [7], SOAP [6], WSDL [12] and UDDI [2]. Each open standard can be brief as follow respectively.

2.1.2 XML (Extensible Markup Language)

The Extensible Markup Language (XML) is a subset of SGML. It was developed by an XML Working Group formed under the auspices of the W3C in 1996. XML describes a class of data objects (called XML documents) and partially describes the behavior of computer programs which process them. The goal of developing XML is to enable generic SGML to be served, received, and processed on the Web in the way that is now possible with HTML. XML has been designed for ease of implementation and for interoperability with both SGML and HTML [7].

XML can be also treated as an application profile or restricted form of SGML (Standard Generalized Markup Language). XML documents are made up of storage units which contain either parsed or unparsed data. Parsed data is made up of characters, some of which form character data, and some of which form markup. Markup encodes a description of the document's storage layout and logical structure. XML provides a mechanism to impose constraints on the storage layout and logical structure. Unparsed data is a resource whose contents may or may not be text, and if text, may be other than XML. Each unparsed data has an associated notation, identified by name. Beyond a requirement that an XML processor make the identifiers for the unparsed data and notation available to the application, XML places no constraints on the contents of unparsed data.

The design goals for XML can be summarized as follow:

1. XML shall be straightforwardly usable over the Internet.
2. XML shall support a wide variety of applications.
3. XML shall be compatible with SGML.
4. It shall be easy to write programs which process XML documents.

5. The number of optional features in XML is to be kept to the absolute minimum.

6. XML documents should be human-legible and reasonably clear.

7. The XML design should be prepared quickly.

8. The design of XML shall be formal and concise.

9. XML documents shall be easy to create.

10. Terseness in XML markup is of minimal importance.

2.1.3 SOAP (Simple Object Access Protocol)

The Simple Object Access Protocol (SOAP) is a lightweight protocol for exchange of information in a decentralized, distributed environment using XML [6]. It defines a simple mechanism for expressing application semantics by providing a variety of systems ranging from messaging system to RPC (Remote Procedure Call). Being an XML based protocol, SOAP consists of three parts: the SOAP Envelope, the SOAP Encoding Rules, and the SOAP RPC Representation. The SOAP Envelope defines a framework for describing what is in a message and how to process it. The SOAP Encoding Rules are used for expressing instances of application-defined data types. The SOAP RPC Representation is a convention for representing remote procedure calls and responses. Also, SOAP can potentially be used in combination with a variety of other protocols.

A major design goal for SOAP is simplicity and extensibility. This means that there are several features from traditional messaging systems and distributed object systems that are not part of the core SOAP specification. Such features include

1. Distributed garbage collection

2. Boxcarring or batching of messages

3. Objects-by-reference (which requires distributed garbage collection)

4. Activation (which requires objects-by-reference)

2.1.4 WSDL (Web Service Description Language)

Web Service Description Language (WSDL) is an XML format for describing network services as a set of endpoints operating on messages containing either document-oriented or procedure-oriented information [12]. The operations and messages are described abstractly, and then bound to a concrete network protocol and message format to define services.

As communications protocols and message formats are standardized in the web community, it becomes increasingly possible and important to be able to describe the communications in some structured way. WSDL addresses this need by defining an XML grammar for describing network services as collections of communication endpoints capable of exchanging messages. WSDL service definitions provide documentation for distributed systems and serve as a recipe for automating the details involved in applications communication.

A WSDL document defines services as collections of network endpoints, or ports. In WSDL, the abstract definition of endpoints and messages is separated from their concrete network deployment or data format bindings. This allows the reuse of abstract definitions:

1. **Messages,** which are abstract descriptions of the data being exchanged.

2. **Port Types**, which are abstract collections of operations.

The concrete protocol and data format specifications for a particular port type constitute a reusable binding. A port is defined by associating a network address with a reusable binding, and a collection of ports define a service. Hence, a WSDL document uses the following elements in the definition of network services:

1. **Types**– a container for data type definitions using some type system (such as XSD).
2. **Message**– an abstract, typed definition of the data being communicated.
3. **Operation**– an abstract description of an action supported by the service.
4. **Port Type**–an abstract set of operations supported by one or more endpoints.
5. **Binding**– a concrete protocol and data format specification for a particular port type.
6. **Port**– a single endpoint defined as a combination of a binding and a network address.
7. **Service**– a collection of related endpoints.

It is important to observe that WSDL does not introduce a new type definition language. WSDL recognizes the need for rich type systems for describing message formats, and supports the XML Schemas specification (XSD) [5] as its canonical type system. However, since it is unreasonable to expect a single type system grammar to be used to describe all message formats present and future, WSDL allows using other type definition languages via extensibility.

2.1.5 UDDI (Universal Description Discovery & Integration)

Universal Description Discovery & Integration (UDDI) is an online Web Service that service consumers can use from web applications to dynamically discover other online services, all neatly packaged in a simple XML interface [2]. In UDDI, service consumers must register for using web services that are provided by service providers so that they can discover the web service information with UDDI searching mechanism.

In general, UDDI is designed as a set of services supporting the description and discovery of:

1. Businesses, Organizations, and other Web services providers.
2. The Web services they make available.
3. The technical interfaces which may be used to access those services.

Based on a common set of industry standards, including HTTP, XML, XML Schema, and SOAP, UDDI provides an interoperable, foundational infrastructure for a Web services-based software environment for both publicly available services and services only exposed internally within an organization. Regarding the data structure of UDDI information, UDDI model is composed of instances of the following entity types:

1. businessEntity: Describes a business or other organization that typically provides Web services.
2. businessService: Describes a collection of related Web services offered by an organization described by a businessEntity.
3. bindingTemplate: Describes the technical information necessary to use a particular Web service.
4. tModel: Describes a "technical model" representing a reusable concept, such as a Web service type, a protocol used by Web services, or a category system.
5. publisherAssertion: Describes, in the view of one businessEntity, the relationship that the businessEntity has with another businessEntity.
6. subscription: Describes a standing request to keep track of changes to the entities described by the subscription.

2.2 QoS-aware Web Service Selection – Existing Solutions

2.2.1 QoS for Web Service

The concept of QoS (Quality of Service) can be applied for any application such as tourism, retail, estate agency, etc. Regarding QoS in web service, Menasce (2002) had addressed the QoS issues for web service [32]. According to his research, QoS is a combination of several qualities or properties of a service, such as availability, security, response time, etc. The QoS measure is observed by web services users. These users are not human beings but programs that send requests for services to web service providers. QoS issues in Web services have to be evaluated from the perspective of the providers of web services and from the perspective of the users of these services.

The main definition about QoS for web service follows the announcement by W3C Working Group Note 25 November 2003 [40] and have had been applied by many researches in recent years [15][32][33][46]. For providing such a better QoS under dynamic and unpredictable characteristics of the web services, it is first necessary to identify all the possible QoS requirements for web services. For this purpose, there were 13 QoS attributes proposed by W3C [40]:

1. **Performance**: The performance of a web service represents how fast a service request can be completed. It can be measured in terms of throughput, response time, latency, execution time, and transaction time, and so on [36][37]. Throughput is the number of web service requests served in a given time interval. Response time is the time required to complete a web service request. Latency is the round-trip delay (RTD) between sending a request and receiving the response. Execution time is the time taken by a web service to process its sequence of activities. Finally, transaction time represents the time that passes while the web service is completing one complete transaction. This transaction time may

depend on the definition of web service transaction.

In general, high quality web services should provide higher throughput, faster response time, lower latency, lower execution time, and faster transaction time.

2. **Reliability**: Web services should be provided with high reliability. Reliability here represents the ability of a web service to perform its required functions under stated conditions for a specified time interval [23]. The reliability is the overall measure of a web service to maintain its service quality. The overall measure of a web service is related to the number of failures per day, week, month, or year. Reliability is also related to the assured and ordered delivery for messages being transmitted and received by service requestors and service providers [36].

3. **Scalability**: Web services should be provided with high scalability. Scalability represents the capability of increasing the computing capacity of service provider's computer system and system's ability to process more users' requests, operations or transactions in a given time interval [37]. It is also related to performance. Web services should be scalable in terms of the number operations or transactions supported.

4. **Capacity**: Web services should be provided with the required capacity. Capacity is the limit of the number of simultaneous requests which should be provided with guaranteed performance [37]. Web services should support the required number of simultaneous connections.

5. **Robustness**: Web services should be provided with high robustness. Robustness here represents the degree to which a web service can function correctly even in the presence of invalid, incomplete or conflicting inputs [37]. Web services should still work even if incomplete parameters are provided to the service request invocation.

6. **Exception Handling**: Web services should be provided with the functionality of

exception handling. Since it is not possible for the service designer to specify all the possible outcomes and alternatives (especially with various special cases and unanticipated possibilities), exceptions should be handled properly [37]. Exception handling is related to how the service handles these exceptions.

7. **Accuracy**: Web services should be provided with high accuracy. Accuracy here is defined as the error rate generated by the web service [37]. The number of errors that the service generates over a time interval should be minimized.

8. **Integrity**: Integrity for web services should be provided so that a system or component can prevent unauthorized access to, or modification of, computer programs or data. There can be two types of integrity: data integrity and transactional integrity. Data integrity defines whether the transferred data is modified in transit. Transactional integrity refers to a procedure or set of procedures, which is guaranteed to preserve database integrity in a transaction [36].

9. **Accessibility**: Web services should be provided with high accessibility. Accessibility here represents whether the web service is capable of serving the client's requests [36]. High accessibility can be achieved, e.g., by building highly scalable systems.

10. **Availability**: The web service should be ready (i.e., available) for immediate consumption. This availability is the probability that the system is up and related to reliability [16]. Time-to-Repair (TTR) is associated with availability. TTR represents the time it takes to repair the web service [36]. The service should be available immediately when it is invoked.

11. **Interoperability**: Web services should be interoperable between the different developmental environments used to implement services so that developers using those services do not have to think about which programming language or operating system the

services are hosted on [36].

12. **Security**: Web services should be provided with the required security. With the increase in the use of web services which are delivered over the public Internet, there is a growing concern about security. The web service provider may apply different approaches and levels of providing security policy depending on the service requestor.

    Security for web services means providing authentication, authorization, confidentiality, traceability/auditability, data encryption, and non-repudiation. Each of these aspects is described below [36][37].

    - Authentication: Users (or other services) who can access service and data should be authenticated.
    - Authorization: Users (or other services) should be authorized so that they only can access the protected services.
    - Confidentiality: Data should be treated properly so that only authorized users (or other services) can access or modify the data.
    - Accountability: The supplier can be hold accountable for their services.
    - Traceability and Auditability: It should be possible to trace the history of a service when a request was serviced.
    - Data encryption: Data should be encrypted.
    - Non-Repudiation: A user cannot deny requesting a service or data after the fact. The service provider needs to ensure these security requirements.

13. **Network-Related QoS Requirements**: To achieve desired QoS for web services, the QoS mechanisms operating at the web service application level must operate together with the QoS mechanisms operating in the transport network (e.g., RSVP, DiffServ, MPLS,

etc.) which are rather independent of the application. In particular, application level QoS parameters should be mapped appropriately to corresponding network level QoS parameters. Basic network level QoS parameters include network delay, delay variation, and packet loss, and they are described as follows.

- The network delay is the average length of time a packet traverses in a network. The network delay can be handled by a good network design that minimizes the number of hops encountered and by the advent of faster switching devices like Layer 3 switches and tag switching system such as MPLS systems and ATM switches.
- The delay variation is the variation in the inter-packet arrival time (leading to gaps, known as jitter, between packets) as introduced by the variable transmission delay over the network. Removing jitter requires collecting packets in buffers and holding them long enough to allow the slowest packets to arrive in time to be played in correct sequence. Jitter buffers may cause additional delay, which is used to remove the packet delay variation as each packet transits the network.
- The Internet does not guarantee delivery of packets. Packets will be dropped under peak loads and during periods of congestion. Approaches used to compensate for packet loss include replay of the last packet, and transmission of redundant information. Out of order packets may need to be re-ordered at the receiver.
- In addition, network management mechanisms may also be involved in controlling and managing QoS for web services.

Even though W3C defines different attributes such as reliability, security, and efficiency as part of web service QoS model, but it leaves the users to judge the level of QoS. This may result in the inconsistency of consumers' views on the values of QoS attributes. That is, one consumer may perceive a particular QoS attribute differently from another consumer.

Therefore, more criteria should be developed based on QoS definition proposed by W3C if it would be taken for developing QoS-aware selection of web service.

2.2.2 Current Literatures about QoS-aware Web Service Selection

A number of QoS-aware web services selection mechanisms have been developed in recent years. These mechanisms focus on performance improvement in order to facilitate web service composition in an open and dynamic environment. They can be briefed as follow.

Menasce (2004) studies the QoS of component web services in terms of cost and execution time [31]. From his view, an internet application can invoke several services, that is, web services composition. As what a stock-trading web service being constructed, for instance, a payment service and an authentication service would be invoked as well. This structure of web service composition makes the web service could be either specified statically or established dynamically. Especially for dynamic composition of web service, the service consumers would be required to discover service providers in terms of functional requirement (cost, facilities, etc.) and nonfunctional requirement in QoS (performance, reliability, etc.).

Therefore, a probability techniques as employed for measuring the both aspects of requirement, including cost and execution time, of component web services by considering different execution scenarios such as parallel, sequential, fastest-predecessor-triggered and so on. This study helps in selecting appropriate component Web services for web service composition. However, it does not consider any consensus from service consumers nor does it take into account QoS attributes with fuzzy definitions. It could be hard to reason why different service consumers would have different behaviors on composed web service selection.

Jaeger, Michael C. (2005) proposed a mechanism for composite web services with pattern-based QoS aggregation [25]. The QoS aggregation is used to verify that a set of services satisfies the QoS requirement for the required composite web services. In this approach the aggregation of QoS for service composition is defined by using a number of pre-defined composition patterns [24] which include QoS ratings. The concept of the composition pattern is inspired by van der Aalst's Workflow Pattern [44]. The identified workflow in web service composition is represented by directed graphs in order to impose the restrictions on the order in which activities are executed specified by the selected aggregation scheme. Based on the model, the aggregation of numerical QoS dimensions is performed and the required web composition is determined and executed.

Similar to issue obtained in Menasce's study (2004), the fuzzy representation of QoS characteristics, which plays an important role in the service selection, should be aware of but had been ignored. This can be improved by introducing fuzzy terms in the representation of the QoS in order to avoid the problem associated with crisp terms. Meanwhile, the combinational pattern-based QoS aggregation could raise very high computation complexity. The criteria and constraints for the pattern-based QoS, if they could be aggregated, would be also difficult to identify.

Furthermore, the issues associated with aggregating different service consumers' fuzzy views on the attributes are not considered. For example, different views on definition of the term "good performance" may exist among service consumers. It is essential to have consistent definitions of these terms for service consumers to discover and select desired services and for a service provider to use such definitions in service advertisement.

Liu, Yutu et al. (2004) proposed an open, fair and dynamic QoS computation model for web service selection. The model is tested using a QoS registry in a hypothetical phone service provisioning market place application [29]. The aim of this model is to investigate the relationship between QoS value and the business criteria, and to study the effectiveness of price and the service sensitivity factors in QoS computation.

This QoS computation model indeed covers both functional and nonfunctional web service composition when it investigates the relationship between QoS value and the business. However, the issue of how to combine different QoS characteristics is not addressed. Different weightings may be given to different characteristics to form a compound request in order to reflect service consumers' preferences. It would obviously impact the relation between QoS value and business criteria because of changed QoS structure. Also, similar to the studies introduced above, the fuzzy representation of QoS should be discussed but was not addressed.

2.2.3 Research Foundation - MFDM

MFDM was proposed by Chi-Chun Lo et al in 2005 for constructing a model which can perform a moderated fuzzy matchmaking for web service [22]. It was proposed to achieve effective web service discovery through a moderated fuzzy matchmaking mechanism. MFDM not only measures the similarity between services in terms of capability, syntax and semantics [22][39][42], but also uses the services' underlying data and information as discovery and selection criteria.

MFDM is built upon fuzzy logic, a semantic web, and decision support methods. In addition it provides a set of procedures for service consumers and providers to follow so that they can reach consensus on the representation of services' contents [8]. A built-in domain dependent fuzzy classifier is employed to classify into concise semantic representation for service discovery, a large amount of data and information stored in services' repositories. The moderation process initiated by a fuzzy moderator minimizes the differences among service consumers. The feedback from consumers on vague queries can be tracked in order to help categorizing similar terms into fuzzy classes.

MFDM consists of a number of system components in following figure can be stated respectively:

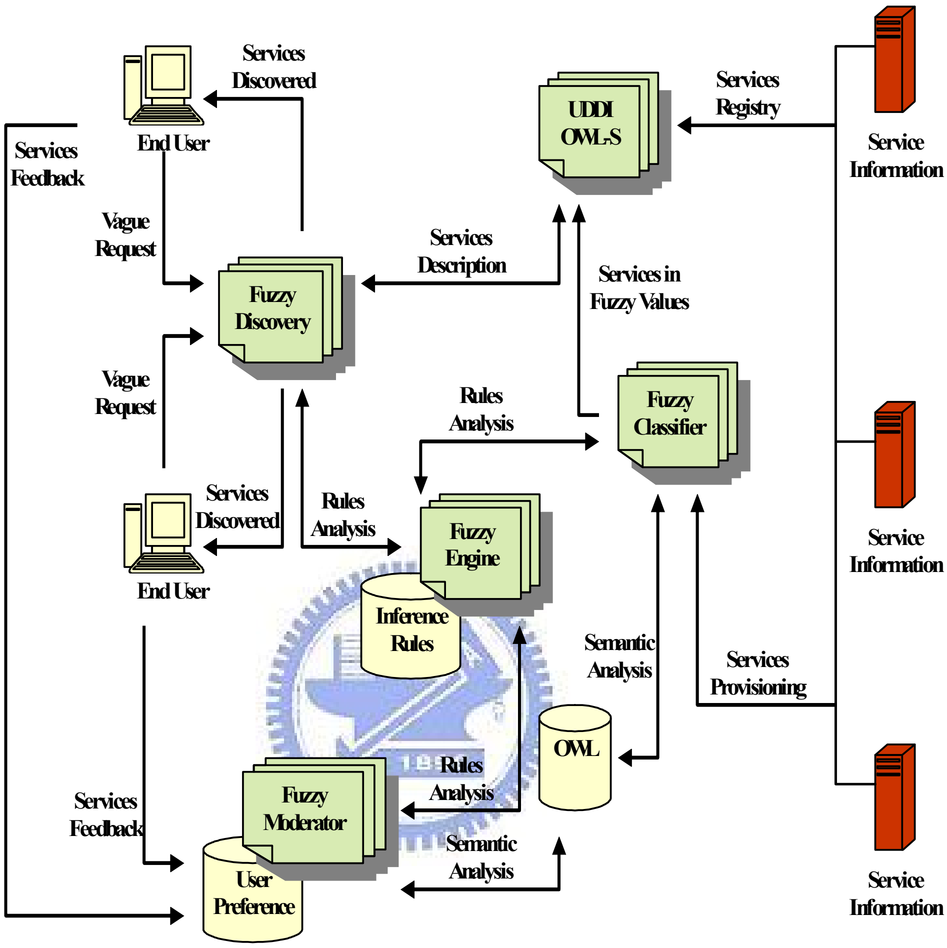

Figure 4: The Framework for the MFDM

According to the framework of MFDM, the fuzzy classifier is able to interpret raw data stored in the service provider's repositories and represent them with fuzzy terms. These fuzzy terms will be employed by the service provider to advertise their services via UDDI. Since UDDI does not have the facility for modeling semantics, the OWL is used for capturing the semantics. The opinions and preferences given by the service providers and consumers are processed via Fuzzy Moderator in order to identify their consensus. This enables service consumers (issuing vague requests) and the service providers (using different terms for service advertisement) to coordinate their expectations.

In the case study for using MFDM, a QoS term "cheap" was taken as an evaluation how the information matchmaking being carried out via similarity aggregation method (SAM). However, the QoS term "cheap" is related to the application it could choose but has nothing to do with the QoS for web service which considers such as reliability, security, interoperability, etc.. That is, QoS for web service had not been fully addressed in MFDM. This was also the motivation why QCMA was developed based on MFDM.

2.3 Multi-Attributes-based Opinions Clustering - Existing Solutions

Regarding multi-attributes-based opinions clustering approach, there are a number of existing solutions developed for this requirement. The amount of the literatures reporting the theoretical developments and their applications are vast [14][17][20][26][27][28][41][43][45]. However, there are three main categories from the literatures research regarding information clustering which are reasoned as very significant to the multi-attributes-based information clustering addressed: shifting or scaling -based clustering, parallel clustering and fuzzy clustering [14][41][45]. In this section, we only briefly describe and analyze the three important literatures that are related to this research. A comprehensive literature review on this area can be found in the [26][45].

Wang, Haixun et al [41] (2003) proposed pCluster model for multi-dimensional pattern similarity clustering in large data sets. In this research, Huixun Wang et al proposed data clustering in term of similarity via correlation between two given multi-attributes-based patterns, which is identified by shifting relationship or scaling relationship, rather than traditional distance-based similarity such as Euclidean distance, Manhattan distance, cosine distance, etc. Therefore, pCluster model is used to cluster "shifting patterns" or "scaling patterns" from large data sets.

E-commerce is the major application for pCluster in the study. In the sample of analyzing 4 types pattern, which was denoted as (*a, b, c, d*), 0 ≦ *a, b, c, d* ≦ 10, (1, 2, 3, 6), (2, 3, 4, 7), (4, 5, 6, 9) were recognized as "similar" by shift comparison in the study. It is indeed can be one way to cluster opinions. However, for patterns such as (1, 2, 3, 2), (2, 3, 4, 3), (7, 8, 9, 8), (8, 9, 10, 9), for instance, obviously (1, 2, 3, 2), (2, 3, 4, 3) could be reasoned as "lower qualified opinions" and (7, 8, 9, 8), (8, 9, 10, 9) could be reasoned as "higher qualified opinions". It implies that "low qualified opinions" could be given in non-serious attitude so that the difference among each type of opinion *a, b, c, d*, could be not really "different". That is, (1, 2, 3, 2), (2, 3, 4, 3) could be given in arbitrary way. Therefore, if (1, 2, 3, 2), (2, 3, 4, 3), (7, 8, 9, 8), (8, 9, 10, 9) will be clustered in same group due to the shift pattern similarity, it could be distorted. To cluster (1, 2, 3, 2), (2, 3, 4, 3), (7, 8, 9, 8), (8, 9, 10, 9) into two groups, {(1, 2, 3, 2), (2, 3, 4, 3)} and {(7, 8, 9, 8), (8, 9, 10, 9)}, should be more reasonable if they can be evaluated by "opinions similarity in perceptional distance" first.

Therefore, the similarity (in perception) between given opinions in the same group should be recognized but is not addressed in pCluster. It makes "similar objects" clustered by pCluster could be not really "similar" due to different perception by web service participants in opinions. Somehow "different perception in opinions" should be the reason why these opinions must be clustered into different groups rather than be clustered together. Also, the weighting problem among all attributes (types) in given object is not discussed. All given attributes should not be treated as "equal weight" due to different significance. On the other hand, for some significant attributes in higher weight, if they are not highlighted in similarity analysis, it could weaken the correlation in similarity between given two objects in comparison.

Fazeli, M. et al [14] (2005) proposed a parallel algorithm tackling multi-features data clustering for multi-computer with star topology in 2005. The proposed parallel algorithm completes in complexity of $O(K+S^2-T^2)$ for a clustering problem of N data patterns with M

features per pattern and K clusters, where $N.M = S!$, $K.M = T!$. In the study, the data to be clustered is depicted with a feature vector v which is a set of measurements (v_1, v_2, …, v_M) map to properties of a collection of data into a Euclidean space of dimension M. It divides N multi-features data into K clusters via specific clustering criteria and the K clusters can be represented as (S_1, S_2, …, S_K) shown as below:

$$S_k = \{i \mid C[i] = k, 0 \leqq k \leqq K-1\} \quad (13)$$

A popular clustering technique, *squared-error algorithm*, is taken for the multi-features data clustering with the square distance $d2$ between pattern i and cluster k shown as below:

$$d2[i,k] = \sum \left(F[i,j] - centre[k,j]\right)^2 \quad (14)$$

Where the cluster centre is obtained by mean of feature matrix $F[i, j]$, which indicates i^{th} data with j^{th} feature, and represented as a ($1 \times M$) vector. With the $|S_k|$ which indicates the cardinality or size of S_k, the centre of cluster k can be defined as below:

$$centre[k,j] = \frac{1}{|S_k|} \sum_{i \in S_k} F[i,j], 0 \le j \le M \quad (15)$$

The *squared-error algorithm* is definitely used to compute the distance $d2[i, k]$ of each pattern i in each cluster k, and choose the minimum distance to all cluster centers. Therefore, all pattern i can be efficiently clustered into right cluster according to minimum distance to corresponded cluster centre. However, even though the multi-features data, similar to multi-QoS attributes opinions, can be clustered via the parallel algorithm, the possible weight distribution among these M dimension (v_1, v_2, …, v_M) should be also discussed but was not addressed, neither. Also, for some data which could not less than the squared distance but very close to the "boundary" could be meaningful if these data on boundary should be treated as "fuzzily similar data" in cluster. However, the fuzzy boundary situation was not discussed. If the "fuzzily similar data" but not less than the squared distance would not clustered, it could make the clustering being distorted.

Definitely, a fuzzy clustering approach which is defined based on consensus in similarity should be considered as the basis to determine the preference of web service selection in multi-dimensional opinion space. Jain, A. K. et al [26] (1999) and Xu, Rui [45] (2005) have made significant review of fuzzy clustering respectively. According to review by A. K. Jain et al, a Fuzzy Clustering Algorithm was introduced with following steps:

1. Select an initial fuzzy partition of the N objects into K clusters by selecting the $N \times K$ membership matrix U. An element u_{ij} of this matrix represents the grade of membership of object x_i in cluster c_j. Typically, u_{ij} is belong to [0,1].

2. Using U, find the value of a fuzzy criteria function, e.g., a weighted squared error criterion function, associated with corresponding partition. One possible fuzzy criteria function is:

$$E^2(\Psi, U) = \sum_{i=1}^{N} \sum_{k=1}^{K} u_{ik} \left\| x_i - c_k \right\|^2 \qquad (16)$$

   where $c_k = \sum_{i=1}^{N} u_{ik} x_i$ is the k^{th} fuzzy cluster centre and Ψ is the pattern sets in U. Reassign patterns to clusters to reduce the criteria's function value and re-compute U.

3. Repeat step 2 until entries in U do not change significantly.

In the fuzzy clustering algorithm, u_{ij} and K should be properly set in preliminary stages. However, the way how to evaluate u_{ik} and K was not addressed. Also, same as previous multi-attributes-based clustering method, the possible weight distribution among these attributes was not addressed, neither.

Xu, Rui [45] (2005) had reviewed a Fuzzy Clustering Method (FCM) which can solve the issue of boundaries among clusters that are not well separated and ambiguous. All selected objects can be clustered into right groups with a certain degree of membership [38]. FCM is the recommended method for fuzzy clustering [20]. FCM attempts to find a partition (c fuzzy clusters) for a set of data points $x_j \in \Re^d$, j = 1 .. N while minimizing the cost function shown as below:

$$J(U,M)=\sum_{i=1}^{c}\sum_{j=1}^{N}(u_{i,j})^m D_{ij} \tag{17}$$

where

1. $U=[u_{i,j}]_{c\times N}$: is the fuzzy partition matrix and $u_{i,j}\in[0,1]$ is the membership coefficient of jth object in the ith cluster.

2. $M = [m_1, m_2, \ldots, m_c]$: is the cluster prototype (mean or cluster) matrix.

3. $m \in [1,\infty)$: is the fuzzy parameter and usually is set to 2 [17].

4. $D_{ij} = D(x_j, m_i)$: is the distance measure between x_j and m_i.

The standard FCM, in which the Euclidean or L_2 norm distance function is used, is summarized as follow:

1. Select appropriate values for m, c, and a small positive number ε. Thereafter, initialize the prototype matrix M randomly and set step variable t = 0.

2. Calculate (at t = 0) or update (t > 0) the membership matrix U by:

$$u_{ij}^{(t+1)}=\frac{1}{\left(\sum_{l=1}^{c}\left(\frac{D_{lj}}{D_{ij}}\right)^{\frac{1}{1-m}}\right)} \qquad \text{for } i = 1, \ldots, c \text{ and } j = 1 \,..\, N. \tag{18}$$

3. Update the prototype matrix M by

$$m_i^{(t+1)}=\frac{\sum_{j=1}^{N}\left(u_{ij}^{t+1}\right)^m x_j}{\left(\sum_{j=1}^{N}\left(u_{ij}^{t+1}\right)^m\right)} \qquad \text{for } i = 1, \ldots, c. \tag{19}$$

4. Repeat steps 2 ~ 3 until $|M^{(t+1)} - M^t| < \varepsilon$.

Besides the initialization on m, c, and ε could be issues in identification, the outlier(s) could appear if m, c, and ε are set with improper value. Also, same as the issues in previous researches, the possible weight distribution among these attributes is not addressed, neither.

Chapter 3. QoS Consensus Moderation Approach (QCMA)

3.1 Approaches, Framework and Behavior

For solving the issues in the literatures regarding QoS-aware selection of web service, QCMA was proposed based on our previous work MFDM [22] for complimenting existing research works by considering service consumers' subjective views and their arbitrary preferences on QoS attributes of web service systems by employing a set of mechanisms to assist them to reach a consensus on QoS attributes in web services. For the objective, QCMA is employed to obtain and moderate group consensus on QoS attributes (such as reliability, performance and interoperability) for web services selection. Comparing with MFDM, it enhances the moderation process on MFDM which focuses on effective web service discovery based on a moderated fuzzy matchmaking mechanism for service inquiries (such as pricing, facilities and some application oriented terms), by improving the method of reaching group opinion similarities and preferences on QoS attributes for web service system.

In QCMA, an initial set of web services and web service consumers' opinions have to be established in order to build a preliminary group consensus. The consumers and providers have to make a judgment on the quality of the participating web services by expressing and defining their subjective opinions such as good reliability, bad performance and high availability etc., on all pre-determined 13 QoS attributes as well as giving their preference ordering over these attributes. The QCMA, including a set of reasoning approaches, is able to analyze and compute the opinions and their preferences to determine a group QoS consensus on these services. So, the QoS of each service can be advertised in UDDI for service discovery and selection according to the reached consensus.

QCMA also provides a moderation mechanism to accommodate the new opinions given from new consumers and new services as well as to reflect the changes from the consumers and users in the dynamic environment. One of the characteristics of QCMA is its flexibility that allows the consumers to express fuzzy QoS opinions. So, the fuzzy QoS opinions from these consumers were analyzed through two phases: group similarity analysis via SAM and QoS preference order analysis via RMGDP. For figuring out what QCMA was organized and how QCMA run QoS-aware selection of web service for unique group framework, besides the theories of SAM and RMGDP, the enhanced functions in QCMA and QCMA system behavior will be described in following sections respectively.

3.1.1 Similarity Aggregation Method (SAM)

SAM was developed for resolving conflicts that emerged from different opinions [9] [10]. In SAM different fuzzy opinions will be aggregated into an opinion consensus class so that they can be measured by their similarities to each other. Therefore, the similarity measuring method is the key to generating the consensus index in the fuzzy opinions set. This characteristic was used by the Fuzzy Moderator for moderating definitions of fuzzy terms. During the process of fuzzy term moderation, the consensus indexes are collected and a consensus agreement is formed. The procedure to perform SAM for QCMA was organized into 8 steps [21]:

1. First, each participant represents his/her subjective fuzzy preference on each specific QoS attributes with a positive trapezoidal fuzzy number, which is denoted as $wsa_{a_i}^{k}$ (wsa : web service activity), as consumer k's fuzzy QoS opinion (k $\in$ K the set of users) on QoS attribute a_i which can be shown as equation (1). The fuzzy trapezoidal number for consumer k's disposition on each QoS attribute can be illustrated as below:

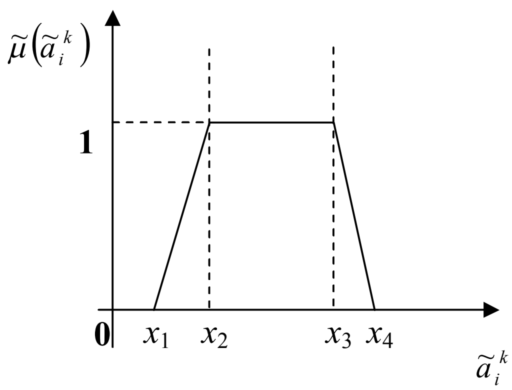

Figure 5: A Trapezoidal Fuzzy Number

In QCMA, the positive trapezoidal fuzzy number is used to describe the fuzzy perception of each QoS attribute. All the fuzzy QoS opinions on each QoS attribute will be collected in the set with $wsa_{a_i}^k$, which is denoted as WSA_{a_i}, for further group-based analysis on opinions similarity and QoS preference. By QoS definition in W3C [40], there are 13 QoS attributes used for evaluating web service QoS, which can be denoted as (a_1, a_2, a_3, …, a_{13}). Therefore, the WSA_{a_i} can be shown as equation (2).

2. This step is to obtain opinion similarity between any two opinions, here the opinion is donated as $wsa_{a_i}^k$ and $wsa_{a_i}^j$, for the specific criterion. The similarity between $wsa_{a_i}^k$ and $wsa_{a_i}^j$, which is denoted as $Sim_{a_i}^{jk}$, can be obtained via the following equation:

$$Sim_{a_i}^{jk} = \frac{\int \left(\min\left\{\tilde{\mu}(wsa_{a_i}^j), \tilde{\mu}(wsa_{a_i}^k)\right\}\right)dx}{\int \left(\max\left\{\tilde{\mu}(wsa_{a_i}^j), \tilde{\mu}(wsa_{a_i}^k)\right\}\right)dx} \tag{20}$$

where $\int \left(\min\left\{\tilde{\mu}(wsa_{a_i}^j), \tilde{\mu}(wsa_{a_i}^k)\right\}\right)dx$ indicates the consistent (overlapped) area between $wsa_{a_i}^j$ and $wsa_{a_i}^k$ which can be depicted as Figure 6,

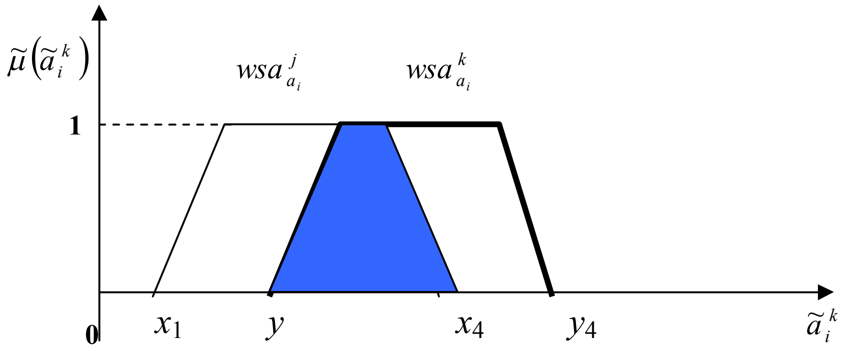

Figure 6: The Consistent Area between two opinions: $wsa_{a_i}^j$ **and** $wsa_{a_i}^k$

and $\int (\max\{\tilde{\mu}(wsa_{a_i}^j), \tilde{\mu}(wsa_{a_i}^k)\})dx$ indicates the total area including $wsa_{a_i}^j$ and $wsa_{a_i}^k$ which can be depicted as Figure 7.

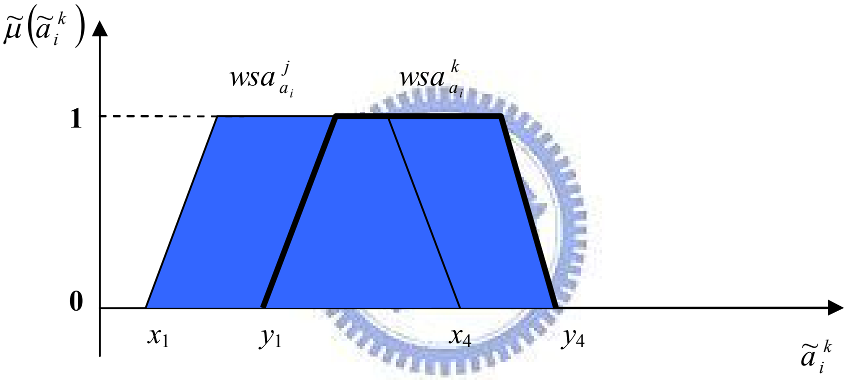

Figure 7: The total area including two opinions: $wsa_{a_i}^j$ **and** $wsa_{a_i}^k$

Although (20) is the definition of similarity used in the original formulation of SAM, it is possible to change this step and use alternative measures of similarity. Such a change does not require alterations to the other steps in the method.

3. This step is to build an *AM* (Agreement Matrix) for each QoS attribute a_i, which can be represented as equation (21), showing each similarity between pairs of opinions in the group.

$$AM_{n\times n} = \begin{bmatrix} 1 & Sim_{a_i}^{12} & \cdots & Sim_{a_i}^{1j} & \cdots & Sim_{a_i}^{1n} \\ Sim_{a_i}^{21} & 1 & \vdots & \vdots & \vdots & \vdots \\ \vdots & \cdots & 1 & \vdots & \vdots & \vdots \\ Sim_{a_i}^{i1} & \cdots & \cdots & 1 & \vdots & Sim_{a_i}^{in} \\ \vdots & \cdots & \cdots & \cdots & 1 & \vdots \\ Sim_{a_i}^{n1} & Sim_{a_i}^{n2} & \cdots & Sim_{a_i}^{nj} & \cdots & 1 \end{bmatrix}_{n\times n} \tag{21}$$

4. This step calculates an AAD (Average Agreement Degree), denoted as $A(wsa_{a_i}^k)$, for each opinion $wsa_{a_i}^k$ in the group. The value of $A(wsa_{a_i}^k)$ can be obtained from equation (22):

$$A(wsa_{a_i}^k) = \frac{1}{n-1}\sum_{\substack{j=1 \\ k\neq j}}^{n} Sim_{kj} \tag{22}$$

5. This step obtains a RAD (Relative Agreement Degree) for each individual opinion using the following formula.

$$RAD(wsa_{a_i}^k) = \frac{A(wsa_{a_i}^k)}{\sum_{j=1}^{n} A(wsa_{a_i}^j)} \tag{23}$$

6. This step involves the assignment of a weighting variable, w_k, to each opinion.

7. This step obtains the CDC (Consensus Degree Coefficient) for each participant:

$$CDC(wsa_{a_i}^k) = \beta \times w_k + (1-\beta) \times RAD(wsa_{a_i}^k) \tag{24}$$

where β is a control variable to indicate the relation between the experts and the unmoderated opinions of the users. All the RAD($wsa_{a_i}^k$) can be obtained through similarity analysis. However, the variation between RAD($wsa_{a_i}^k$) and CDC($wsa_{a_i}^k$) would be quite smaller for large population of users no matter what β would be set as (It can be verified in the FMG-QCMA Validation and Evaluation). Therefore, it is possible to simplify the use of CDC($wsa_{a_i}^k$) by setting β in CDC($wsa_{a_i}^k$) as zero so that CDC($wsa_{a_i}^k$) is equal to RAD($wsa_{a_i}^k$).

8. Aggregate the fuzzy opinions by the CDC in (24) as the formula as below:

$$\widetilde{R}_{a_i} = \sum_{k=1}^{n} CDC(wsa_{a_i}^k) \times wsa_{a_i}^k \tag{25}$$

where $\widetilde{R}_{a_i}$ indicates an "overall" fuzzy number of combining all opinions on QoS attribute a_i.

The eighth steps in SAM defined by Hsu, His-Mei and Chen, Chen-Tung [21]. Deriving from MFDM, the eighth step which aggregates the fuzzy opinions by CDC of each opinion from service participant for reaching group consensus was not used in QCMA due to its characteristic of unique group framework. Therefore, in QCMA only 7 steps (step 1 to step 7) were used for identifying similarity in group which was the foundation of preference order analysis via RMGDP.

3.1.2 Resolution Method for Group Decision Problems (RMGDP)

Opinion similarity enables the service consumers to reach a consensus on the interpretation of a specific QoS attribute for web services. However, among a number of QoS attributes in web service, the different preferences on these attributes must be thought over. The preference on different QoS attribute cannot be told via recognizing the opinions similarity on one of the specific QoS attribute. The preference on different QoS attribute must be further realized even though the similarity for fuzzy QoS opinions on each QoS attribute can be reasoned as well.

It is reason why RMGDP (Resolution Method for Group Decision Problems) is adopted for the series of studies about web service selection. In QCMA and FMG-QCMA, RMGDP was proposed to alleviate their differences on preferences. In RMGDP, three steps: The Transformation Phase, The Aggregation Phase and The Exploitation Phase, construct the whole process and will be illustrated respectively in following sections.

The Transformation Phase:

In the transformation phase, all participants will be grouped. Each participant has to evaluate alternatives according to given criteria, and to assign his/her preference orders to the related alternatives.

The participants allocate their preference ordering based on subjective judgments. The position of alternative a_i (QoS attribute) for participant k is denoted o_i^k. A transfer function, p_{ij}^k, is defined for converting these relative positions of alternatives to a preference relation which sets an ordering preference degree relating alternatives a_i and a_j. For $wsa_{a_i}^k$:

$$p_{ij}^k = f(o_i^k, o_j^k) = \frac{1}{2}\left(1 + \frac{o_j^k}{m-1} - \frac{o_i^k}{m-1}\right) \tag{26}$$

where p_{ij}^k is a preference relation given by $wsa_{a_i}^k$ based on the relative positions o_i^k, o_i^k for attributes a_i and a_j respectively. m indicates the number of alternatives (attributes) in the analysis.

The Aggregation Phase:

In the aggregation phase, p_{ij}^c is defined by aggregating the participants' preferences $\{p_{ij}^1, \ldots, p_{ij}^n\}$ for a particular pair i, j by means of a fuzzy majority [9]. In QCMA and FMG-QCMA, the fuzzy majority is formed with the OWA (Ordered Weighted Averaging) operator F_Q and the fuzzy quantifier Q. The function with F_Q and Q aggregates the individual preference values to obtain the group preference order from n users via the following formula:

$$p_{ij}^c = F_Q(p_{ij}^1, \ldots, p_{ij}^n) = \sum_{i=1}^{n} w_i b_i \tag{27}$$

where b_i is the i-th large value in $\left(p_{ij}^1, p_{ij}^2, \cdots, p_{ij}^n\right)$ and $w_i = Q(i/n) - Q((i-1)/n)$. The values of $Q(i/n)$ are determined by the particular fuzzy quantifier used.

The Exploitation Phase:

This phase calculates the consequence of collecting each alternative priority into group preferences. Two well-known fuzzy ranking methods, Quantifier guided Non-Dominance Degree (*QGNDD*) and Quantifier guided Dominance Degree (*QGDD*) [35] are adopted to provide different aspects for the evaluation of alternative priorities.

QGNDD is a fuzzy ranking method based on fuzzy preference relations. The method determines the relative preference degree of the alternatives. The Non-Dominance Degree (*NDD*) fuzzy ranking can be calculated from the participants' group preference relations, and is formulated as follows:

$$(u_{NDD})_{ji} = 1 - \max\{p_{ji}^c - p_{ij}^c, 0\} \tag{28}$$

A membership function $u_{NDD}(a_i)$ based on Eq. (28) can be interpreted as the degree to which a_i is not dominated by any other a_j ($j = 1, \ldots, m$). The function $u_{NDD}(a_i)$ is taken to find the highest order of alternatives. The *NDD* for alternative a_i is taken to identify a criterion which has a higher preference degree than others. For a linguistic quantifier Q (e.g. "most"), the *NDD* of the linguistic quantifier is represented as *QGNDD* defined as below:

$$QGNDD(a_i) = F_Q(1 - d_{ji}^c, j = 1 \ldots m, j \neq i) \tag{29}$$

where $d_{ji}^c = \max\{p_{ji}^c - p_{ij}^c, 0\}$. According to (27), (29) can be represented as:

$$F_Q(1-d_{1i}^c, 1-d_{2i}^c, \ldots, 1-d_{mi}^c) = w_1.(1-b_1) + \ldots + w_m(1-b_m) \qquad (30)$$

where b_i is the i-th smallest value in $\left(d_{1i}^c, d_{2i}^c, \cdots, d_{mi}^c\right)$

The solution offered by equations. (29) (30) indicates that the fuzzy majority in the remaining alternatives a_j (j = 1, …, m) cannot dominate the alternative a_i. Also, all the preferences in the alternatives can be prioritized and the corresponding order can be obtained.

QGNDD cannot be used for ordering of the preferences if $u_{NDD}(a_i)$ obtained from numerous alternatives is in an Unfuzzy Nondominated (*UND*) situation [35], i.e., $u_{NDD}(a_i) = 1$. Also, in order to avoid more than two *UND* situations occurring simultaneously, the obtained fuzzy preference orders need to be validated by other fuzzy ranking methods such as *QGDD*.

Using equation (26) *QGDD* can quantify the dominance for each a_i which has preference order over all other alternatives and used for prioritizing the final ordering preference. Its values are given using the following equation:

$$QGDD(a_i) = F_Q(p_{ij}^c, j = 1 \ldots m, i \neq j) = \sum_{i=1}^{m} w_i b_i \qquad (31)$$

where $w_i = Q(i/m) - Q((i-1)/m)$ and b_i is the i-th largest value in $\left(p_{i1}^c, p_{i2}^c, \cdots, p_{im}^c\right)$. By (31) the *UND* situation can be resolved and final preference ranking for $a_1, a_2, \ldots, a_m$ can be determined.

3.1.3 QCMA Functions Enhancement and Behavior

Deriving from MFDM, QCMA extension uses QoS evaluation for consensus analysis and moderation. For reaching the QoS-aware selection of web service, QCMA includes additional components (which include *Quali-Fuzzy Discoverer*, *Quali-Fuzzy Engine*, UDDI / OWLS, *Quali-Fuzzy Classifier*, *Fuzzy Moderator* and *QoS Fuzzy Moderator*) in order to improve the functional enhancement for QoS moderation. Figure 8 represents QCMA components which are explained as follows.

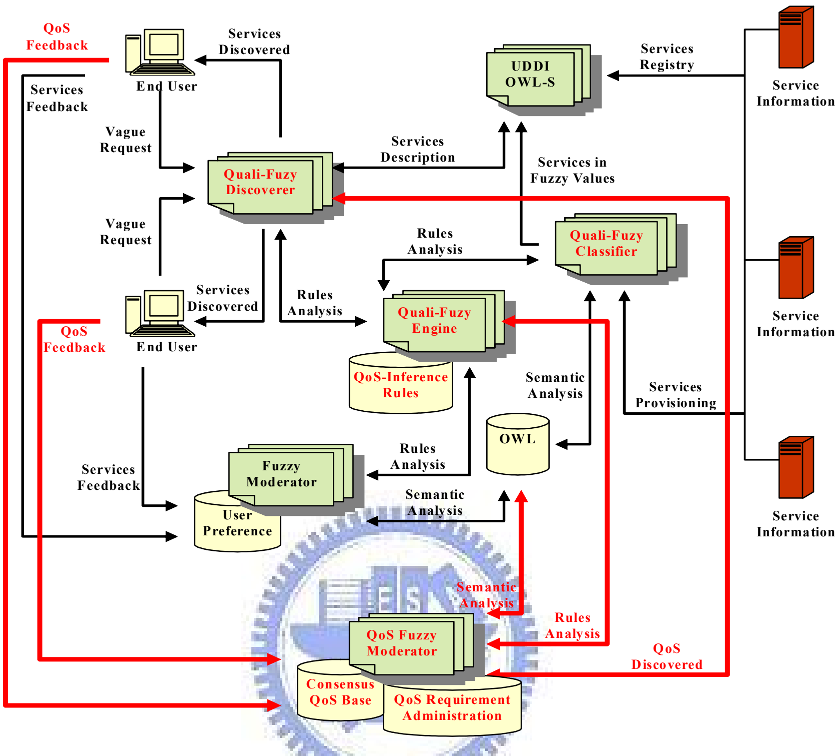

Figure 8: The Framework of QCMA

Each component in QCMA derived from MFDM can be described as follow:

1. *UDDI / OWLS:* The service information and corresponding QoS fuzzy attributes and their associated definitions are deployed to UDDI and represented in OWLS before the discovery process takes place. Since there is no record input from service consumers when the system is initialized, the QoS attributes (through QoS Fuzzy Moderator) will be created in UDDI and the initial values for the QoS terms will be assigned by the service providers. The update to QoS attributes will be executed when the consumers start to have their feedback.

2. *Quali-Fuzzy Classifier:* classifies and interprets all participating web service information that includes all corresponding QoS fuzzy attributes. The web service information

provided by web service providers will be classified by the Quali-Fuzy Classifier according to the representation of the fuzzy QoS attributes. In addition, the vague requests issued by service consumers will be analyzed semantically by Quali-Fuzy Discoverer and to be forwarded to and interpreted by Quali-Fuzy Classfier according to its fuzzy classification. The process includes the discovery of web service information and possible QoS requirements with the help of the Quali-Fuzy Engine.

However, the classification rules are modeled in OWL. The rules are triggered and reasoned over the domain information in order to produce the required knowledge for OWLS and UDDI. When Quali-Fuzy Classifier receives the request from Quali-Fuzy Engine, the meanings of the given fuzzy terms and expected QoS can be interpreted. As a result, the related information can be retrieved from UDDI / OWLS using a pattern match.

3. *Quali-Fuzy Engine:* is designed to analyze the vague inquiry and the QoS requirements received from Quali-Fuzy Discoverer. After receiving the input from Quali-Fuzy Discoverer, Quali-Fuzy Engine reasons over the input with Fuzzy logic and interprets the fuzzy terms in the request which have been processed by Quali-Fuzy Classifier. If both fuzzy terms and the corresponding QoS expectation need to be tuned after rule analysis, Quali-Fuzy Engine will either communicate with QoS Fuzzy Moderator for fuzzy terms modification, or communicate with QoS Fuzzy Moderator for QoS modification.

4. *Quali-Fuzy Discoverer:* receives all vague requests from service consumers for the selection of the appropriate services. Quali-Fuzy Discoverer receives vague request (including possible given QoS requirement) and requests the feedback from the users' perceptions and opinions on QoS in order to modify service definition after locating and selecting the required web services. The steps involved not only analyzing the semantic definition of each vague request, but also examining the meaning of the required quality attribute which is represented in the vague request. Quali-Fuzy Discoverer intensifies the

intelligence of Fuzzy Discovery and supports meaningful and concise discovered web information analysis through either Quali-Fuzy Engine or UDDI / OWLS.

5. *QoS Fuzzy Moderator:* is dedicated to tune both QoS terms and QoS perception derived from service consumers which are associated with corresponding web service information deployed in UDDI / OWLS. In the system initialization, QoS Requirement Administration provides an initial set of QoS term definitions for group consensus.

In order to reach the group consensus on the definitions of QoS terms, the service consumers' subjective opinions and preferences over QoS have to be registered and stored in QoS Requirement Administration in advance of further processing. When additional service consumers with different opinions or preferences join, the process of moderating group consensus may have to take place. So, any new opinions or requests have to be analyzed by comparing with the information in the QoS Consensus Base in order to determine whether the moderation process has to be carried our or not.

According to the above description, the service consumers will first register their QoS expectations (definitions) with QoS Requirement Administration. For example, a service consumer may demand a query regarding service performance by specifying the condition: "The response time should not be slow". Using fuzzy analysis this condition can be interpreted as "the response time delay should be no more than 7~10 seconds"). All the fuzzy terms with corresponding QoS representations used by the service providers have been employed in UDDI and declared in QoS Requirement Administration (via Quali-Fuzy Classifier). Since UDDI does not have facility for modeling semantics, the OWL is used for capturing the semantics.

With the availability of the required information provided by service consumers and providers, the Quali-Fuzy Classifier (including built-in domain knowledge) is able to interpret

QoS information within services and represent them in a fuzzy way. The fuzzy information will be employed by the service provider to advertise their service via UDDI. The opinions, with expected QoS requirements and preferences, given by the service consumers, will be processed via both the Fuzzy Moderator and the QoS Fuzzy Moderator in order to identify their consensus. This enables service consumers (issuing vague requests with QoS representation) and the service providers (using QoS fuzzy terms for service advertisement) to moderate their expectations.

This opinion $wsa_{a_i}^{k}$ can be also treated as an input to QoS Fuzzy Moderator at the moderation stage that involves SAM and RMGDP. However, in order to cluster the opinions of the service consumers who have similar opinions on QoS in QCMA, the threshold values for CDC in SAM for QoS attribute a_i, which are donated as $t_{a_i}^{cdc\_l}$ and $t_{a_i}^{cdc\_u}$, are set to correspond all $wsa_{a_i}^{k}$ which can be classified into the same $S_{cdc\_l}^{cdc\_u}(wsa_{a_i}^{k})$. $S_{cdc\_l}^{cdc\_u}(wsa_{a_i}^{k})$ indicates the fuzzy QoS opinion set which can be defined as below:

$$S_{cdc\_l}^{cdc\_u}(wsa_{a_i}^{k}) = \left\{ wsa_{a_i}^{k} \mid t_{a_i}^{cdc\_l} \le CDC(wsa_{a_i}^{k}) \le t_{a_i}^{cdc\_u} \right\} \quad (32)$$

All the RAD($wsa_{a_i}^{k}$) can be obtained through similarity analysis. To simplify the operation of CDC($wsa_{a_i}^{k}$), we set β in CDC($wsa_{a_i}^{k}$) as zero so that CDC($wsa_{a_i}^{k}$) is equal to RAD($wsa_{a_i}^{k}$). In other words, all $wsa_{a_i}^{k}$ with "similar" relative agreement degree was made for grouping them into $S_{cdc\_l}^{cdc\_u}(wsa_{a_i}^{k})$.

After the SAM process is completed, each $wsa_{a_i}^{k}$ in $S_{cdc\_l}^{cdc\_u}(wsa_{a_i}^{k})$ having consistent definitions over the QoS terms will be used for preference analysis via RMGDP. The preference order of QoS terms for each $wsa_{a_i}^{k}$ and the group preference of QoS terms for $S_{cdc\_l}^{cdc\_u}(wsa_{a_i}^{k})$ will be ranked. Since $wsa_{a_i}^{k}$ comprises the fuzzy web service terms given by service consumers, the result generated by SAM and RMGDP will be treated as the consequence of the QoS group consensus.

3.2 Validation and Evaluation

For validating how QCMA can be helpful for unique group-based consensus in web service selection, a prototype system based on a case study using hotel booking web services was developed in order to validate the functionality of the proposed approach.

3.2.1 Similarity Analysis via SAM in QCMA

In the validation and evaluation for QCMA, there are 50 QoS opinions randomly generated by simulation system shown in Appendix A (A.1 and A.2). Each QoS opinion ($wsa_{a_i}^{k}$) for the consumer k will be represented as trapezoidal fuzzy number defined in (1). However, for simplifying the process of handling fuzzy QoS opinion, the $(x_1)_{a_i}^{k}$ and $(x_2)_{a_i}^{k}$ in each $wsa_{a_i}^{k}$ are set as null so that each $wsa_{a_i}^{k}$ can be shown as figure as below:

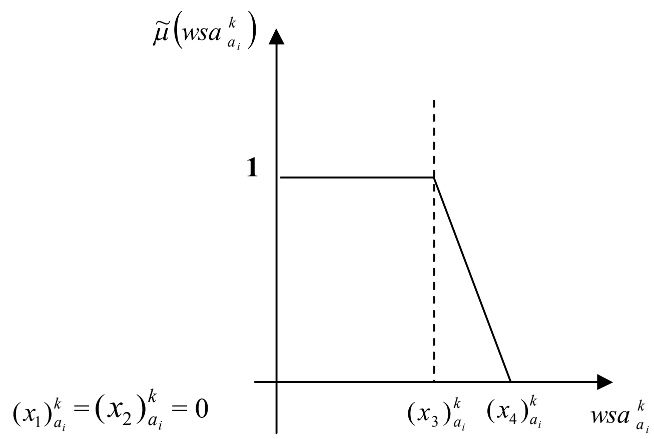

Figure 9: The Representation of $wsa_{a_i}^{k}$ in Validation for QCMA

For instance, the service consumer k considers "Acceptable Reliability" as (4 ~ 7) shown in Appendix A (A.1 and A.2). This can be mapped to QoS attribute (a_2) in $wsa_{a_2}^{k}$ which can be defined as below:

0 : 30% reliability	1 : 40% reliability
2 : 50% reliability	3 : 60% reliability
4 : 70% reliability	5 : 80% reliability
6 : 85% reliability	7 : 90% reliability
8 : 95% reliability	9 : 98% reliability
10 : 100% reliability	

Figure 10: The Scale of QoS "Reliability"

Therefore, for the degree pattern of (4 ~ 7), $wsa_{a_2}^{k}$ can be represented as:

$$wsa_{a_2}^{k} \leftarrow (0,0,4,7) \tag{33}$$

Since each individual consumer's fuzzy definition over the QoS term has been obtained, the similarity between each pair of feedback from all $wsa_{a_i}^{k}$ can be analyzed via SAM and thirteen agreement matrixes (for all service consumers) for thirteen QoS attributes in S_Q can be generated as below:

$$AM_{a_1} = \begin{bmatrix} 1 & 0.93 & \cdots & 0.87 & \cdots & 0.88 \\ 0.93 & 1 & \vdots & \vdots & \vdots & 0.82 \\ \vdots & \cdots & 1 & \vdots & \vdots & \vdots \\ 0.87 & \cdots & \cdots & 1 & \vdots & 0.76 \\ \vdots & \cdots & \cdots & \cdots & 1 & \vdots \\ 0.88 & 0.82 & \cdots & 0.76 & \cdots & 1 \end{bmatrix}_{50\times 50} \tag{34}$$

....

....

....

$$AM_{a_{13}} = \begin{bmatrix} 1 & 0.73 & \cdots & 0.93 & \cdots & 0.93 \\ 0.73 & 1 & \vdots & \vdots & \vdots & 0.79 \\ \vdots & \cdots & 1 & \vdots & \vdots & \vdots \\ 0.93 & \cdots & \cdots & 1 & \vdots & 1.0 \\ \vdots & \cdots & \cdots & \cdots & 1 & \vdots \\ 0.93 & 0.79 & \cdots & 1.0 & \cdots & 1 \end{bmatrix}_{50\times 50}$$

By analyzing similarity between each pair of feedback from all $wsa_{a_i}^{k}$, we can obtain RAD and individual CDC for each service consumer which is shown as below.

$$A(Consumer_{01}) = \{.89,.90,.80,.91,.85,.92,.91,.82,.85,.71,.91,.91,.81\}$$

$$RAD(Consumer_{01}) = \{.021,.020,.019,.019,.022,.020,.020,.019,.019,.017,.021,.022,.019\}$$

$$CDC(Consumer_{01}) = \{.021,.020,.019,.019,.022,.020,.020,.019,.019,.017,.021,.022,.019\}$$

....

.... (35)

....

$$A(Consumer_{50}) = \{.82,.90,.86,.87,.88,.92,.93,.88,.92,.88,.90,.84,.86\}$$

$$RAD(Consumer_{50}) = \{.019,.020,.020,.020,.022,.021,.020,.020,.021,.021,.021,.020,.020\}$$

$$CDC(Consumer_{50}) = \{.019,.020,.020,.020,.022,.021,.020,.020,.021,.021,.021,.020,.020\}$$

The result of RAD is the same as the one produced by CDC in this experiment since β was set as zero. Let each $t_{a_i}^{cdc\_l}$ is set as 0.025 and $t_{a_i}^{cdc\_u}$ is set as 0.015 (i = 1 .. 13) in order to verify if all QoS feedback can be treated as “similar”. As a result, we can conclude that the consumers have shared similar opinions on the definitions of QoS terms. Therefore, the fifty $wsa_{a_i}^{hbs(k)}$ (k = 1 ~ 50) were treated and classified into a group consensus.

3.2.2 Preference Analysis via RMGDP

RMGDP is employed to identify the possible compromised preference order from their diverse preferences. In this case study the order preference for the fifty $wsa_{a_i}^{hbs(k)}$ (k = 1 ~ 50) is set as shown as Appendix A (A.3).

In Appendix A (A.3), these order preference for the fifty $wsa_{a_i}^{hbs(k)}$ are denoted as ($o_{hbs}^{1}, o_{hbs}^{2}, o_{hbs}^{3}$ ….., o_{hbs}^{50}) and each o_{hbs}^{k} can be represented as follow:

$$
\begin{aligned}
o_{hbs}^{1} &= \{a_8, a_3, a_7, a_4, a_1, a_9, a_{11}, a_6, a_5, a_2, a_{12}, a_{13}, a_{10}\} \\
o_{hbs}^{2} &= \{a_3, a_8, a_4, a_7, a_{11}, a_5, a_1, a_6, a_9, a_{12}, a_2, a_{13}, a_{10}\} \\
&\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\
o_{hbs}^{50} &= \{a_8, a_{13}, a_{11}, a_9, a_6, a_7, a_5, a_3, a_1, a_4, a_2, a_{12}, a_{10}\}
\end{aligned}
\qquad (36)
$$

Using the ($o_{hbs}^{1}, o_{hbs}^{2}, o_{hbs}^{3}$ ….., o_{hbs}^{50}), the (p^{1}, p^{2}, p^{3} ……, p^{50}) can be obtained via transformation phase of RMGDP as below:

$$
p^{1} = \begin{bmatrix}
.50 & .71 & .38 & .46 & .67 & .63 & .42 & .33 & .54 & .83 & .58 & .75 & .79 \\
.29 & .50 & .17 & .25 & .46 & .42 & .21 & .13 & .33 & .63 & .38 & .54 & .58 \\
.63 & .83 & .50 & .58 & .79 & .75 & .54 & .46 & .67 & .96 & .71 & .88 & .92 \\
.54 & .75 & .42 & .50 & .71 & .67 & .46 & .38 & .58 & .88 & .63 & .79 & .83 \\
.33 & .54 & .21 & .29 & .50 & .46 & .25 & .17 & .38 & .67 & .42 & .58 & .63 \\
.38 & .58 & .25 & .33 & .54 & .50 & .29 & .21 & .42 & .71 & .46 & .63 & .67 \\
.58 & .79 & .46 & .54 & .75 & .71 & .50 & .42 & .63 & .92 & .67 & .83 & .88 \\
.67 & .88 & .54 & .63 & .83 & .79 & .58 & .50 & .71 & 1.0 & .75 & .92 & .96 \\
.46 & .67 & .33 & .42 & .63 & .58 & .38 & .29 & .50 & .79 & .54 & .71 & .75 \\
.17 & .38 & .04 & .13 & .33 & .29 & .08 & .00 & .21 & .50 & .25 & .42 & .46 \\
.42 & .63 & .29 & .38 & .58 & .54 & .33 & .25 & .46 & .75 & .50 & .67 & .71 \\
.25 & .46 & .13 & .21 & .42 & .38 & .17 & .08 & .29 & .58 & .33 & .50 & .54 \\
.21 & .42 & .08 & .17 & .38 & .33 & .13 & .04 & .25 & .54 & .29 & .46 & .50
\end{bmatrix}
$$

$$
\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \qquad (37)
$$

$$
\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots
$$

$$
p^{50} = \begin{bmatrix}
.50 & .58 & .46 & .54 & .42 & .33 & .38 & .17 & .29 & .67 & .25 & .63 & .21 \\
.42 & .50 & .38 & .46 & .33 & .25 & .29 & .08 & .21 & .58 & .17 & .54 & .13 \\
.54 & .63 & .50 & .58 & .46 & .38 & .42 & .21 & .33 & .71 & .29 & .67 & .25 \\
.46 & .54 & .42 & .50 & .38 & .29 & .33 & .13 & .25 & .63 & .21 & .58 & .17 \\
.58 & .67 & .54 & .63 & .50 & .42 & .46 & .25 & .38 & .75 & .33 & .71 & .29 \\
.67 & .75 & .63 & .71 & .58 & .50 & .54 & .33 & .46 & .83 & .42 & .79 & .38 \\
.63 & .71 & .58 & .67 & .54 & .46 & .50 & .29 & .42 & .79 & .38 & .75 & .33 \\
.83 & .92 & .79 & .88 & .75 & .67 & .71 & .50 & .63 & 1.0 & .58 & .96 & .54 \\
.71 & .79 & .67 & .75 & .63 & .54 & .58 & .38 & .50 & .88 & .46 & .83 & .42 \\
.33 & .42 & .29 & .38 & .25 & .17 & .21 & .00 & .13 & .50 & .08 & .46 & .04 \\
.75 & .83 & .71 & .79 & .67 & .58 & .63 & .42 & .54 & .92 & .50 & .88 & .46 \\
.38 & .46 & .33 & .42 & .29 & .21 & .25 & .04 & .17 & .54 & .13 & .50 & .08 \\
.79 & .88 & .75 & .83 & .71 & .63 & .67 & .46 & .58 & .96 & .54 & .92 & .50
\end{bmatrix}
$$

As the default value set in QoS Fuzzy Moderator, the initial weight value w_i in equation (27) will be set as:

$$
w_i = 0.02, \ (i = 1 \ldots 50) \qquad (38)
$$

Therefore, the collective preference p^c can be obtained as:

$$p^c = \begin{bmatrix} .50 & .55 & .41 & .44 & .49 & .43 & .38 & .39 & .45 & .49 & .47 & .56 & .59 \\ .45 & .50 & .36 & .39 & .44 & .38 & .34 & .34 & .40 & .44 & .43 & .51 & .54 \\ .59 & .64 & .50 & .53 & .57 & .52 & .47 & .47 & .53 & .58 & .56 & .65 & .67 \\ .56 & .61 & .47 & .50 & .55 & .49 & .44 & .45 & .51 & .55 & .53 & .62 & .65 \\ .52 & .56 & .43 & .46 & .50 & .44 & .40 & .40 & .46 & .51 & .49 & .58 & .60 \\ .57 & .62 & .49 & .51 & .56 & .50 & .46 & .46 & .52 & .56 & .55 & .64 & .66 \\ .62 & .67 & .53 & .56 & .60 & .54 & .50 & .50 & .56 & .61 & .59 & .68 & .70 \\ .61 & .66 & .53 & .55 & .60 & .54 & .50 & .50 & .56 & .60 & .59 & .68 & .70 \\ .56 & .60 & .47 & .50 & .54 & .48 & .44 & .44 & .50 & .55 & .53 & .62 & .64 \\ .51 & .56 & .42 & .45 & .50 & .44 & .39 & .40 & .46 & .50 & .48 & .57 & .60 \\ .53 & .58 & .44 & .47 & .51 & .45 & .41 & .41 & .47 & .52 & .50 & .59 & .61 \\ .44 & .49 & .35 & .38 & .42 & .37 & .32 & .32 & .38 & .43 & .41 & .50 & .52 \\ .41 & .46 & .33 & .35 & .40 & .34 & .30 & .30 & .36 & .40 & .39 & .48 & .50 \end{bmatrix} \quad (39)$$

According to equation (29)(30), the QGNDD for each QoS attribute can be represented as below:

$$\begin{aligned} QGNDD(a_1) &= F_Q\left(1-d^c_{02\_01}, 1-d^c_{03\_01}, \cdots, 1-d^c_{13\_01}\right) \\ QGNDD(a_2) &= F_Q\left(1-d^c_{01\_02}, 1-d^c_{03\_02}, \cdots, 1-d^c_{13\_02}\right) \\ &\cdots\cdots \\ QGNDD(a_{13}) &= F_Q\left(1-d^c_{01\_13}, 1-d^c_{02\_13}, \cdots, 1-d^c_{13\_12}\right) \end{aligned} \quad (40)$$

According to equation (31), the QGNDD for each QoS attribute can be represented as below:

$$\begin{aligned} QGDD(a_1) &= F_Q\left(p^c_{01\_02}, p^c_{01\_03}, \cdots, p^c_{01\_13}\right) \\ QGDD(a_2) &= F_Q\left(p^c_{02\_01}, p^c_{02\_03}, \cdots, p^c_{02\_13}\right) \\ &\cdots\cdots \\ QGDD(a_{13}) &= F_Q\left(p^c_{13\_01}, p^c_{13\_02}, \cdots, p^c_{13\_12}\right) \end{aligned} \quad (41)$$

According to equation (29)(30)(31), the initial weight value w_i for each b_i in QoS Fuzzy Moderator will be set as 0.083. This demonstrates that QGNDD and QGDD for all QoS attributes can be fairly assessed without any bias.

Therefore, the evaluation for thirteen QoS attributes via both QGNDD and QGDD can be represented in Table 1:

Table 1: The QoS Preference Order for Hotel Booking Web Service by QCMA

QoS Attribute	QGNDD		QGDD
	Evaluation	UND Occurs	Evaluation
a_1	0.907	a_2, a_{12}, a_{13}	0.471
a_2	0.827	a_{12}, a_{13}	0.418
a_3	0.990	a_1, a_2, a_4, a_5, a_8, a_9, a_{10}, a_{11}, a_{12}, a_{13}	0.565
a_4	0.975	a_1, a_2, a_4, a_8, a_9, a_{10}, a_{11}, a_{12}, a_{13}	0.536
a_5	0.928	a_1, a_2, a_{10}, a_{12}, a_{13}	0.488
a_6	0.984	a_1, a_2, a_4, a_5, a_9, a_{10}, a_{11}, a_{12}, a_{13}	0.550
a_7	1.000	All the other attributes	0.597
a_8	1.000	All the other attributes	0.593
a_9	0.971	a_1, a_2, a_5, a_{10}, a_{11}, a_{12}, a_{13}	0.531
a_{10}	0.922	a_1, a_2, a_{12}, a_{13}	0.482
a_{11}	0.942	a_1, a_2, a_5, a_{10}, a_{12}, a_{13}	0.499
a_{12}	0.793	a_{13}	0.403
a_{13}	0.753	No UND Occurs	0.377

In this case, a number of UND situations occur in the QGNDD analysis shown in the following result.

$$o_{hbs}^{c} = \{a_7 = a_8, a_3, a_6, a_4, a_9, a_{11}, a_5, a_{10}, a_1, a_2, a_{12}, a_{13}\} \tag{42}$$

The result shows that the consensus preference for a_7 is the same as a_8. The QGNDD analysis may not be able to produce complete order of the preferences. The auxiliary method, QGDD, is deployed to identify the complete order of consensus preferences. For the cases of a_7 vs. a_8, the preference order for QoS consensus via QGDD analysis in o_{hbs}^{c} is:

$$o_{hbs}^{c} = \{a_7, a_8, a_3, a_6, a_4, a_9, a_{11}, a_5, a_{10}, a_1, a_2, a_{12}, a_{13}\} \tag{43}$$

According to the opinions and preferences from fifty participated service consumers, the consensus of group preference order of QoS in hotel booking web service will be:

Accuracy > Integrity > Scalability > Exception Handling > Capacity >
Accessibility > Interoperability > Robustness > Availability > Performance > (44)
Reliability > Security > Network-Related QoS Requirement: (User Friendly)

The representation of the web service information can be organized according to the order of QoS attributes. The obtained order of preference helps QCMA perform more effective web service selection.

3.3 Review on QCMA

The work of developing QCMA focused on the QoS-based web services selection under unique grouping. It performed QoS consensus and to alleviate the differences on QoS characteristics in the web services selection. The proposed QCMA possesses the following features.

1. QCMA is a web service selection mechanism based on fuzzy QoS consensus for a group of participants. The architecture allows them to reach QoS consensus by including a number of activities such as participants' opinion similarity, QoS term preference ordering and QoS fuzzy scale for each QoS term. The contribution of QCMA not only includes the fuzzy inquiry for service selection, but also offers the features to model the QoS preference consensus after aggregating sufficient fuzzy QoS opinions.
2. QCMA is designed for open and dynamic web environment, such that new opinions and preferences as well as new QoS aspects can be modeled flexibly.

Even if the issues above would be resolved in this research, through further research there still have some challenges raised as follow:

1. If the fuzzy QoS opinions were collected from web service participants with very different backgrounds and potentially diverse perceptions, the obtained consensus may not be effective. It makes "opinion group" could be too diverse to build "consensus" in web service selection.
2. Even though the group consensus built upon opinion similarity and QoS preference order

in QCMA under unique group framework, there is still no criterion how QoS similarity and preference order can be combined together for consensus analysis.

3. Some outliers might be re-classified into other appropriate groups if a multi-groups approach is adopted, as these outlier opinions may have meaningful correlation. The omission of those outliers without further examination can be inappropriate. Furthermore, due to multi-attributes structure, these outliers could be identified by different attribute values that are too far away from the consensus. It makes multi-attribute-based outliers identification more difficult to obtain than is the case with single-attribute-based outlier identification.

4. Through SAM in QCMA, the operation complexity which is due to comparing similarity on each pair of QoS opinions is still relative high. With n QoS opinions the complexity will be $O(n^2)$. It is not efficient if the n will be a large number. The operation overhead impacted by the number of n should be relieved especially the n QoS opinions should be treated as a very large data set due to very high population in internet world.

Definitely multi-groups-based consensus for web service selection will be the right structure for solving issues above. The higher similarity, more precise group consensus / corresponded preference order over QoS attributes, and more efficient calculation to handle n QoS opinions can be carried out under multi-groups-based framework. However, multi-attributes-based clustering is much complicated than single-attribute-based clustering. It makes the necessity that an effective / efficient clustering approach for multi-attributes / multi-groups-based QoS-aware selection of web service should be evolved. Therefore, some works which should focus on the investigation of other intelligent approaches should be developed. It was the motivation why FMG-QCMA was developed on second stage of QoS-aware selection of web service.

Chapter 4. Fuzzy Multi-Groups-based QCMA (FMG-QCMA)

4.1 Approaches, Framework and Behavior

FMG-QCMA (Fuzzy Multi-Groups-based QCMA) was evolved from QCMA for carrying out multi-groups-based QoS-aware selection of web service. Differing from QCMA which analyzes the fuzzy opinions and preferences given by the service consumers and providers on a collection of pre-determined web services QoS in attempt to reach consensus on these subjective terms and their preference orders for web service selection, FMG-QCMA is capable of clustering service consumers (fuzzy opinions) into a number of sub-groups according to consumers' similar disposition on pre-determined web services QoS attributes and focuses on the assessment of a specific collection of recommended web services for each clustered sub-group according to its sub-group preference over QoS attributes.

For ensuring the reliability of operation in FMG-QCMA, the service consumers' dispositions in QoS are supposed as relative static over a period of time. Once the consumers' dispositions in QoS are obtained, the service providers supporting various levels of QoS can promote the right quality level of services to the right group of service consumers. When a service request is issued by a service consumer, the service providers will look up the service consumer's profile and provide close match services according to the consumer's past selection patterns and disposition in QoS.

Evolving from QCMA and resolving some issues in current research addressed in literature review, FMG-QCMA was required to works out the challenges for multi-groups-based QoS-aware selection of web service which are listed as follow:

1. Associated weight on each QoS attribute should be identified due to different preference orders given by the service consumers. Therefore, a weighted multi-attributes QoS

similarity should be defined.

2. To prevent from unintentionally removing possible meaningful data which just falls outside of the pre-defined group boundaries, the multi-attributes-based clustering criteria should be formulated with fuzzy evaluation.

3. The complexity (efficiency) of handling multi-groups-based QoS-aware selection of web service should be more efficient than what obtained from unique group-based QoS-aware selection of web service.

4. The accuracy of handling multi-groups-based QoS-aware selection of web service, in similarity, should be better than what obtained from unique group-based QoS-aware selection of web service.

Because of objective to cluster "similar fuzzy QoS opinions" into same sub-group, whether these similar fuzzy QoS opinions are clustered "into" a sub-group or "on" the sub-group fuzzy boundary, the threshold for similarity will be set as fuzzy interval. Also, any fuzzy QoS opinion on a certain sub-group's fuzzy boundary could be also allocated into the other sub-group if this fuzzy QoS opinion is also "similar to" the corresponded sub-group centre. This phenomenon could make these sub-groups having fuzzy overlapping.

The challenges for clustering multi-attributes-based QoS opinions are much higher than traditional single dimensional clustering schemes. The dynamic weight distribution over all QoS attributes for each fuzzy QoS opinion must be thought over. The similarity threshold, even though it can be initialized by expert's advice, could be inappropriate so that it should be moderated by service consumers' later feedback regarding perception in QoS. Also, the similarity for fuzzy QoS opinions and system performance under multi-groups framework should be better than single group-based QCMA.

4.1.1 System Functions Deployment in FMG-QCMA

FMG-QCMA is built with a number of system components derived from QCMA. Those system components existing in QCMA, which include *Quali-Fuzy Classifier*, *UDDI OWL-S*, *Quali-Fuzy Engine* and *Quali-Fuzy Moderator*, are evolved and replaced by *FMQ Distributor*, *FMQ UDDI OWL-S*, *FMQ Engine*, *FMQ Discoverer* and *FMQ Moderator* that can be depicted as follow:

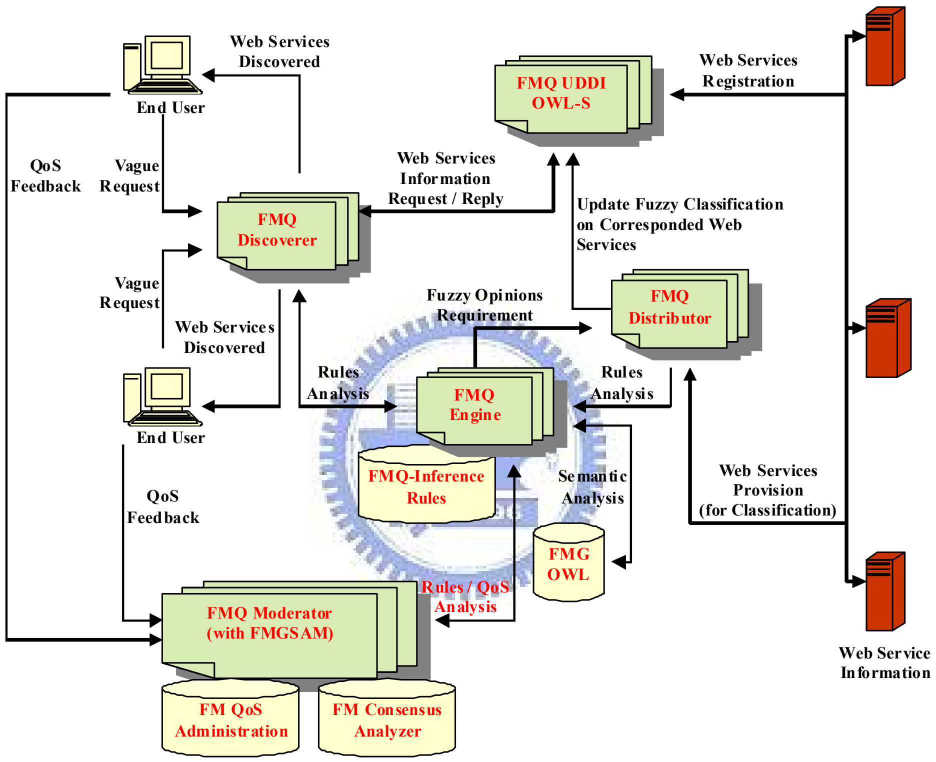

Figure 11: The Framework of FMG-QCMA

1. *FMQ Distributor*: enhances the capability of *Quali-Fuzy Classifier* in QCMA with following functions:

   (1) All collected web service registered in the *FMQ UDDI / OWL-S* will be classified fuzzily according to fuzzy web service management performed in the *FMQ Distributor*. The rule of fuzzy classification on given web service will be analyzed by *FMQ Engine*.

(2) Interpreting fuzzy web service inquiry issued from *FMQ Engine*, the *FMQ Distributor* reasons the fuzzy web services, retrieves the required web service stored in *FMQ UDDI / OWL-S*, returns the required web service back to *FMQ Engine*, and updates the correlated QoS status stored in *FMQ UDDI / OWL-S* in FMG-QCMA.

2. *FMQ UDDI / OWL-S*: is derived from *UDDI / OWL-S* in QCMA for registering and storing the web service which is provided from web service providers (vendors). Besides the web service registration, there are two major operations designed for fuzzy web service handling and corresponded classification:

(1) The fuzzy classification for registered web service, which is updated in *FMQ UDDI / OWL-S*, will be moderated by *FMQ Distributor* by event (driven by analysis from *FMQ Engine*).

(2) The definite web service exploration from service consumers will be performed via *FMQ Discoverer*. Any well defined requests from service consumers will be issued from *FMQ Discoverer* and being dispatched to *FMQ UDDI / OWL-S* directly rather than fuzzily analyzed through *FMQ Engine*, from viewpoint of *FMQ UDDI / OWL-S*.

3. *FMQ Engine*: extends the capability of the *Quali-Fuzzy Engine* in QCMA with the following functions:

(1) *FMQ Engine* analyzes the vague inquiry or the fuzzy QoS opinions (when service consumers set his/her disposition on each QoS attribute and preference order over QoS attributes) received from the *FMQ Discoverer* and reasons over the vague inquiry using fuzzy logic. The rules to interpret the vague inquiry from *FMQ Discoverer* are stored in object *FMQ-Inference Rules*.

(2) *FMQ Engine* ascertains to which fuzzy QoS opinion sub-group the user making the inquiry to *FMQ Discoverer* belongs, QoS attribute reasoning in similarity and QoS

attributes preference order via invoking QoS analysis in *FMQ Moderator*.

(3) *FMQ Engine* asks for retrieving the fuzzily classified web service managed by *FMQ Distributor* by inquiry from *FMQ Discoverer*. The recommended web services for the inquiry will be returned to *FMQ Discoverer* after *FMQ Distributor* replies to *FMQ Engine*.

(4) *FMQ Engine* helps to fuzzily classify web service that *FMQ Distributor* gains from web service providers. The semantic analysis for the request of fuzzy classification will be performed via invoking the rules defined in *FMG OWL*.

4. *FMQ Discoverer*: is the object of Man-Machine interface which handles web service inquiries and fuzzy QoS opinions from service consumers and recommends right web services accordingly. The major operations designed in *FMQ Discoverer* including:

(1) *FMQ Discoverer* receiving all vague requests (fuzzy inquiry or setting of fuzzy QoS opinions) from service consumers for the selection of the appropriate / recommended web services, completely same as what *Quali-Fuzy Discoverer* did for QCMA. Definitely, the vague requests will be converted as fuzzy requirement which will be delivered to *FMQ Engine* for further rule analysis. However, if the requests from service consumers are decoded as well defined requests rather than vague requests, then the "well defined requests" will be converted as a definite inquiry and delivered to *FMQ UDDI / OWL-S* for looking up the web service directly.

(2) When *FMQ Discoverer* receives vague request (including vague inquiries or fuzzy QoS opinions that could be issued by service consumers), it will also request the later feedback from the service consumers' perceptions and opinions on QoS in order to modify service definition after locating and selecting the required web services. The steps involved not only analyzing the semantic definition of each vague request, but

also examining the meaning of the required quality attribute which is represented in the vague request.

5. *FMQ Moderator*: is to improve the capability of the *Quali-Fuzzy Moderator* in QCMA, especially for multi-groups framework of QoS consensus analysis, by including the following functions:

   (1) For the fuzzy QoS opinions *FMQ Moderator* moderates the perception derived from service consumers for the potentially recommended web services deployed in *FMQ UDDI / OWLS*.

   (2) *FMQ Moderator* initializes the FMGSAM operations, including the clustering of all fuzzy QoS opinions. All fuzzy QoS opinions and their temporary analyzed matrixes are stored in *FM Consensus Analyzer*.

   (3) Via FMGSAM operations defined in *FMQ Moderator*, the AM, AAD, RAD, CDC and group consensus in FMGSAM for each collected / converted fuzzy QoS opinion are obtained.

   (4) Verify the later feedback from web service consumers if his / her delivered fuzzy QoS opinion was clustered in right sub-group or not. If the number of later feedbacks for the fuzzy QoS opinions clustering reaches the threshold *m_threshold_distortion* through algorithm *Clustering_Verification*, then the fuzzy QoS opinions clustering will be identified as "Mismatched Similarity" and the whole WSA_{S_Q} will be re-clustered via algorithm *Fuzzy_Clustering* under the condition.

   (5) To perform RMGDP for each clustered sub-group. The result of analyzing preference order over all QoS attributes for each QoS opinion sub-group can be obtained. The outcome of RMGDP will be delivered to *FMQ Engine* for further update on

mechanism of generating recommended web services.

(6) The *FM QoS Administration* provides the set of QoS attribute definitions and initial value of system parameters. In order to reach the consensus on the definitions of QoS attributes for grouped fuzzy QoS opinion of each sub-group, the service consumers' subjective opinions and preferences over QoS attributes have to be registered and stored by *FM QoS Administration*.

4.1.2 System Behavior of FMG-QCMA

In FMG-QCMA, each service consumer needs to express his/her dispositions on all 13 QoS attributes [40] with selection from a set of pre-defined scales and their associated trapezoidal fuzzy number as well as his/her preference order over these QoS attributes. For the objective each of the thirteen QoS attributes is possible to find a numerical measure of quality in the context of the type of service required. The values of this measure can then be scaled to correspond to numbers in the range [0,10]. For each service consumer there will be a range of values that will be considered appropriate for the service they require. At the lower end there will be a cut off value and services with lower values will not be considered in any circumstance. At the higher end there will be a value above which improvement in quality will not be relevant to their needs and services above the threshold will only be considered if they do not cost any extra. So for each attribute a service consumer must choose four points in the range of values.

1. Below this level a service cannot to be considered in any circumstances.
2. This is the lower end of the normal expected quality for a service
3. This is the upper end of the normal expected quality for a service.
4. Getting above this level could not be used to justify extra investment.

Given the choice of these values for each attribute by a service consumer a corresponding set of trapezoidal numbers over the standardized scale of [0,10] can be defined. For instance, for the attribute, performance, the natural measure is a response time in seconds. The upper limit of quality is immediate response, 0 seconds (standardized quality value = 10). The lower limit is context dependent but assuming a straight single retrieval requirement, 10 seconds is taken as the lower limit (standardized quality value = 0). The standardization scaling can most conveniently be presented as a table showing measures corresponding to the eleven scaled values [0, 1,.., 9, 10]. This is shown in Table 2. With reliability the natural quality measure is the percentage of transactions that will be completely successful. The scaling is shown in Table 3.

Table 2: The Natural Measure for Performance Quality

Performance Quality Rating	Response time in Seconds	Performance Quality Rating	Response time in Seconds
0.0	*10.00*	5.5	*1.75*
0.5	*8.00*	6.0	*1.50*
1.0	*7.00*	6.5	*1.25*
1.5	*6.00*	7.0	*1.00*
2.0	*5.00*	7.5	*0.75*
2.5	*4.00*	8.0	*0.50*
3.0	*3.00*	8.5	*0.25*
3.5	*2.75*	9.0	*0.05*
4.0	*2.50*	9.5	*0.02*
4.5	*2.25*	10	*0.00*
5.0	*2.00*		

Table 3: The Natural Measure for Reliability Quality

Reliability Quality Rating	Percentage Transaction Success	Reliability Quality Rating	Percentage Transaction Success
0.0	*50.0%*	5.5	*91.5%*
0.5	*60.0%*	6.0	*92.5%*
1.0	*70.0%*	6.5	*94.0%*
1.5	*72.5%*	7.0	*95.0%*
2.0	*75.0%*	7.5	*97.0%*
2.5	*77.5%*	8.0	*99.0%*
3.0	*80.0%*	8.5	*99.3%*
3.5	*82.5%*	9.0	*99.5%*
4.0	*85.0%*	9.5	*99.8%*
4.5	*87.5%*	10	*100%*
5.0	*90.0%*		

There are similar tables for each of the thirteen attributes which can be presented to service consumers for their choice of the four key levels.

FMG-QCMA, then, can collect these fuzzy QoS opinions to proceed the following four phases of FMG-QCMA operations which is depicted in the following Figure.

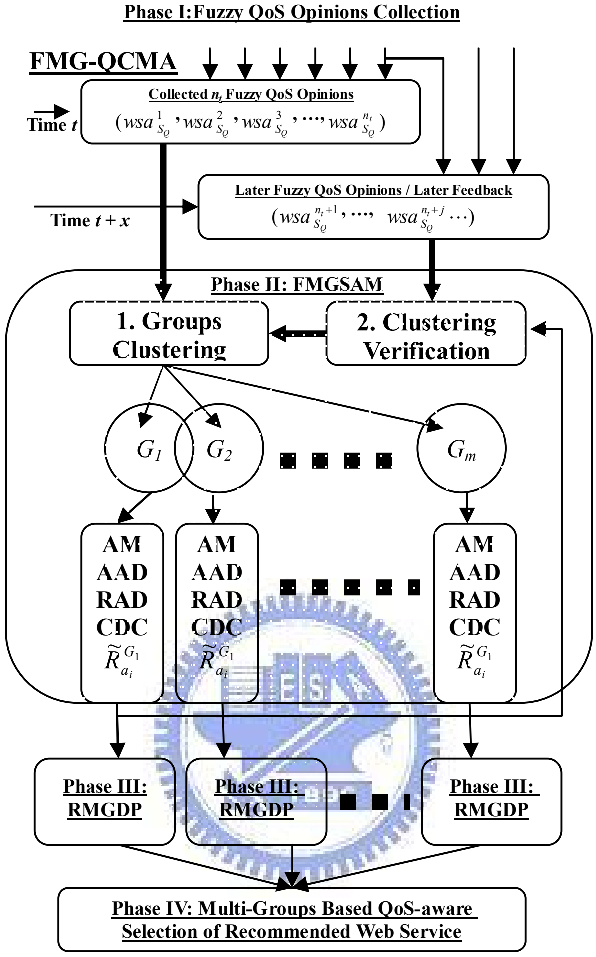

Figure 12: FMG-QCMA System Behavior

In Figure 12, there are four phases for handling all incoming fuzzy QoS opinions:

Phase I: To collect consumers' fuzzy QoS opinions which reflect consumers' disposition in QoS, their preferences order over QoS attributes, and initializing parameters for grouping such as similarity thresholds for any pair of fuzzy QoS opinions and sub-groups' fuzzy boundaries. These values of system parameters will be evaluated by the system so they can be changed or adjusted at later stages, if they are inappropriate.

Phase II: To cluster all the collected fuzzy QoS opinions into sub-groups via *Groups*

Clustering. Each allocation will be evaluated via *Clustering Verification*.

The operation *Groups Clustering* realized by *Algorithm Fuzzy_Clustering* is to populate sub-groups with the collected fuzzy QoS opinions according to measurement of fuzzy QoS opinion similarity. Each fuzzy QoS opinion can be allocated into one or two sub-groups, as it depends on the degree of similarity to the related (close) sub-groups and the pre-set fuzzy boundaries. Each sub-group will be reasoned by using Agreement Matrix (AM) / Average Agreement Degree (AAD) and Relative Agreement Degree (RAD) / Consensus Degree Coefficient (CDC) in order to find the group similarity on fuzzy QoS opinions. In other words, it examines the degree of group consensus over the concept of disposition on the pre-defined QoS attributes.

The operation *Clustering Verification* is performed by *Algorithm Clustering Verification* and used for performing an analysis on new fuzzy QoS opinions from new web service consumers or misallocated existing opinions. In *Clustering Verification*, two scenarios will possibly occur:

1. There are two categories of similarities defined in the system: full and partial membership. Each opinion sub-group has fuzzy boundaries. Two neighboring sub-groups are likely to have overlapping areas which members belong to both groups with different degrees. When a member has full membership to a group, it means that the opinion has been assigned to the right group. The process for allocating this opinion will stop. However, if an opinion has been evaluated as partial membership to a group, it will be evaluated against adjacent group in order to identify its degree of membership. These opinions can be preliminarily clustered into arbitrary number of groups. So, producing good quality in grouping in the first instance is not expected. However, the system can evaluate

the quality by measuring group similarity co-efficiency by using FMGSAM. If it does not reach desired level, the system boundaries or number of groups will be changed accordingly. This process will be iterated until the satisfactory results produced or it could be terminated after a number of tries. All the new fuzzy QoS opinions will have to be explored and analyzed to ensure that they are classified appropriately.

2. The purpose of grouping and identifying consensus on the QoS attributes is to recommend right services to the consumers. If the consumers are often not satisfied with the services recommended by the system, this could be derived from inappropriate settings for the group boundaries or changes of consumer's pattern on service usage. We assume that the consumer will inform the changes. Another set of process will be activated to resolve the issue which will not be discussed here. For the other cases, the system records the events and accumulates these incidents. When the unsatisfactory number reaches or grows beyond the pre-set threshold, then the fuzzy boundaries for the sub-groups will be adjusted in order to improve the accuracy of recommending the appropriate services to the consumer to select. When this occurs, all fuzzy QoS opinions will be re-clustered into new opinion sub-groups.

Phase III: Once the quality of grouping presents a satisfactory result, the preference order for each sub-group can be calculated and obtained via RMGDP.

Phase IV: Through FMGSAM in Phase II and RMGDP in Phase III, the system is ready for use. Since the service consumer group consensus on QoS profiles and their preference orders can be obtained, the service providers can advertise and provide their services according to their targeting groups. The service consumer issuing the request to the system will receive a list of recommended web services which QoS can satisfy the required fuzzy opinions. Another filtering process based on individual QoS preference ordering will be applied in order to reduce unqualified services. After these processes, the consumers can select the desired services.

The proposed multi-attributes and multi-groups service selection is expected to produce better result than the single opinion group approach for service selection. The members in a sub-group should be correlated closer than the single group. The system should be able to recommend close match services to the requests issued by the service consumer. Since the single group has been divided into a number of sub-groups and the size of each sub-group is smaller than or equal to the single group, the computational complexity can be reduced and the system efficiency can be improved. The following gives more detailed descriptions of the key steps in FMG-QCMA.

4.1.3 FMGSAM and Multi-Groups RMGDP

The proposed FMGSAM is designed for similarity analysis under multi-groups framework. Following the system behavior of FMG-QCMA in the previous section, the FMGSAM can be organized with seven steps.

1. Represent All Fuzzy QoS Opinions: Based on the $wsa_{S_Q}^{k}$ represented in (1), the multi-attributes based fuzzy QoS opinion from web service participant *k*, $wsa_{S_Q}^{k}$, is represented for all QoS attributes defined in S_Q, the set of QoS terms in W3C[40], as shown in (3). The set of all the collected fuzzy QoS opinions $wsa_{S_Q}^{k}$, which is donated as WSA_{S_Q}, can be defined as (4).

2. There are two conditions to use operations: *Groups Clustering* or *Clustering Verification*, in this step.

Condition to Use "Groups Clustering"

The operation *Groups Clustering* is activated by either pre-set system time (as time t) or the event of "re-clustering" from the operation *Clustering Verification*. When the operation *Groups Clustering* commences, all collected $wsa^{k}_{S_Q}$ in WSA_{S_Q} will be clustered into appropriate groups (G_1, G_2, …, G_m) through *Algorithm Fuzzy_Clustering* (See Appendix B)and *Algorithm SimVerifier* (See Appendix C) based on the similarity threshold $\tilde{d}_{S_Q}$ and the multi-attribute based similarity $Sim^{jk}_{S_Q}$ between selected $wsa^{j}_{S_Q}$ and $wsa^{k}_{S_Q}$. $Sim^{jk}_{S_Q}$ can be obtained by equation (8). $Sim^{jk}_{a_i}$ indicates the similarity between $wsa^{j}_{a_i}$ and $wsa^{k}_{a_i}$ on QoS attribute a_i and can be obtained by equation (7). It can be noted that this measure of the similarity of two trapezoidal numbers is not the same as (20). This chosen formula is easier to calculate and gives comparable results. The element $so^{jk}_{a_i}$ indicates the similarity of preference order between $o^{j}_{a_i}$ and $o^{k}_{a_i}$ and it can be obtained for the q QoS attributes by equation (9) (by W3C [5], $q = 13$).

Each $Sim^{jk}_{a_i}$ will be compared with the similarity threshold, $\tilde{d}_{S_Q}$, through the operators such as $\tilde{\geq}$, $\tilde{>}$, $\tilde{\leq}$, $\tilde{<}$ and $\cong$ that are defined in *Algorithm SimVerifier*. The pairs of values for $\tilde{d}_{S_Q}$ determine the ways in which the individual similarities can influence the overall similarity. The clustering process requires the contributions to be added together and the total will determine the inclusion, semi-inclusion or exclusion of consumer from a cluster. The thresholds applied to the totals are the values $f_{c\_S_Q}$. (See Appendix C steps 9, 11, 13, 15 and 17)

Condition to Use "Clustering Verification"

The operation *Clustering Verification* (See Appendix D) is launched by the addition of fuzzy QoS opinions contributed from the new web service consumers or by new feedback (mismatch) on unsatisfactory web services recommended by the system. Each new set of fuzzy QoS opinions will be assessed and assigned to appropriate opinion sub-groups if it is either similar to (with full membership) or nearly similar to (with partial membership). (*E_Fail_CDC, E_Fuz_*Sim or *E_Abs_Sim* in Appendix D).

If the threshold of "re-clustering all fuzzy QoS opinions" is reached due to too many mismatch cases or the sub-group opinion consensus coefficient is too low, then a "re-clustering" event will be triggered to activate the operation *Groups Clustering* and this will moderate the threshold (boundaries) of subgroups in order to re-cluster the opinions.

3. Determine Agreement Matrixes $(AM_{p-a_i})_{n_p \times n_p}$ for each clustered opinion group G_p. In the construction of the clusters all the necessary similarities (8) that are need to form the agreement matrices shown in step 3 of SAM have been calculated.

$$(AM_{1-a_1}) = \begin{bmatrix} 1 & Sim_{a_1}^{12} & \cdots & Sim_{a_1}^{1j} & \cdots & Sim_{a_1}^{1n_1} \\ Sim_{a_1}^{21} & 1 & \vdots & \vdots & \vdots & \vdots \\ \vdots & \cdots & 1 & \vdots & \vdots & \vdots \\ Sim_{a_1}^{j1} & \cdots & \cdots & 1 & \vdots & Sim_{a_1}^{jn_1} \\ \vdots & \cdots & \cdots & \cdots & 1 & \vdots \\ Sim_{a_1}^{n_1 1} & Sim_{a_1}^{n_1 2} & \cdots & Sim_{a_1}^{n_1 j} & \cdots & 1 \end{bmatrix}_{n_1 \times n_1} \quad (AM_{1-a_2}) = \begin{bmatrix} 1 & Sim_{a_2}^{12} & \cdots & Sim_{a_2}^{1j} & \cdots & Sim_{a_2}^{1n_2} \\ Sim_{a_2}^{21} & 1 & \vdots & \vdots & \vdots & \vdots \\ \vdots & \cdots & 1 & \vdots & \vdots & \vdots \\ Sim_{a_2}^{j1} & \cdots & \cdots & 1 & \vdots & Sim_{a_2}^{jn_2} \\ \vdots & \cdots & \cdots & \cdots & 1 & \vdots \\ Sim_{a_2}^{n_2 1} & Sim_{a_2}^{n_2 2} & \cdots & Sim_{a_2}^{n_2 j} & \cdots & 1 \end{bmatrix}_{n_2 \times n_2}$$

$$\cdots \quad (AM_{m-a_{13}}) = \begin{bmatrix} 1 & Sim_{a_{13}}^{12} & \cdots & Sim_{a_{13}}^{1j} & \cdots & Sim_{a_{13}}^{1n_m} \\ Sim_{a_{13}}^{21} & 1 & \vdots & \vdots & \vdots & \vdots \\ \vdots & \cdots & 1 & \vdots & \vdots & \vdots \\ Sim_{S_Q}^{j1} & \cdots & \cdots & 1 & \vdots & Sim_{a_{13}}^{jn_m} \\ \vdots & \cdots & \cdots & \cdots & 1 & \vdots \\ Sim_{a_{13}}^{n_m 1} & Sim_{a_{13}}^{n_m 2} & \cdots & Sim_{a_{13}}^{n_m j} & \cdots & 1 \end{bmatrix}_{n_m \times n_m}$$

Figure 13: Agreement Matrixes

4. Determine the Average Agreement Degrees: As in step 4 of SAM (definition (22)) it is possible to find the average agreement degree for each clustered opinion sub-group.

5. Determine the Relative Agreement Degrees: The RAD values within the clusters for each of the customers can be found using step 5 of the SAM process as (23).

6. Determine the Consensus Degree Coefficients: As shown in step 6 of the SAM process it

is possible to moderate the purely customer defined RAD values using weightings $w_{k(G_p)}$ for each $wsa_{S_Q}^{k(G_p)}$ in opinion sub-group G_p. With $w_{k(G_p)}$ and assigned β the CDC for $wsa_{S_Q}^{k(G_p)}$ can be obtained using definition (24).

If the value of CDC is less than the pre-defined threshold, the group boundaries will be adjusted in order to increase group consensus coefficients. For other cases, the system progresses to the next step. This criterion is for the self-assessment mechanism to improve the quality of grouping.

7. If it is necessary definition (25) of the SAM process can be used to provide a consensus trapezoidal numbers for the clusters.

To provide a more detailed analysis of the clusters of customers it is useful to find consensus values for their preferences. The clusters identify similarities of quality expectations and the preferences will show the group's attitudes to the relative importance of these expectations. Therefore, all the QoS opinions $wsa_{S_Q}^{G_p(k)}$ in G_p will be further analyzed via RMGDP according to associated preference order over all QoS attributes. All clustered opinion sub-group (G_1, G_2, …, G_m) there are m RMGDP processes will be performed respectively. In the FMGSAM, the individual consumer's preference ordering over QoS attributes was taken into consideration when the sub-groups are forming. Therefore at this stage, the members in a group should have strong consensus on the preference ordering.

4.1.4 Precision and Efficiency

Calculating similarity for each pair of fuzzy QoS opinions in a group, is the dominant step in the complexity of FMG-QCMA and QCMA frameworks. The improvement on this step without compromising the precision of measurement of opinion similarities can significantly improve system efficiency. In the QCMA, the number of opinions in a single group is n, so its complexity is $O(n^2)$ for AM generation in SAM. The number of processes and its associated complexity in FMG-QCMA can be significantly reduced, as it has multiple opinion sub-groups to fabricate (AM_1), (AM_2) ……, and (AM_m) giving complexity of the form $O(n_1^2) + O(n_2^2) + \ldots + O(n_m^2)$ and this will be lower than $O(n^2)$ since $n= n_1+ n_2+\ldots+ n_m$.

In addition, FMG-QCMA can improve the precision in opinion similarity measurement which is illustrated as following steps:

1. Let $PSim_{FMQ}$ is denoted as precision (lowest similarity) for FMG-QCMA which is obtained from minimal $Sim_{S_Q}^{G_p j}$ in generated (AM_1), (AM_2) ……, (AM_m) defined in Figure 13. Also, let $PSim_Q$ is denoted as precision (lowest similarity) for QCMA which is obtained from minimal $Sim_{S_Q}^{jk}$ in generated AM defined in (21).

2. The precision improvement by FMG-QCMA which compares with QCMA in similarity can be defined as $PImpr$(FMG-QCMA / QCMA):

$$PImpr(\text{FMG-QCMA/QCMA})=(PSim_{FMQ} / PSim_Q)-1 \tag{45}$$

The example below, where it is feasible to calculate the full set of similarities and the similarities used in FMG-QCMA, shows the improvement in precision introduced by the method.

4.2 Validation and Evaluation

This section presents how the proposed FMG-QCMA achieves marketing web services via a case study, hotel booking web services. There were sixty fuzzy QoS dispositions collected from sixty consumers at time t as initial inputs to FMG-QCMA. This output from FMG-QCMA process contains a number of opinion sub-groups. Based on the framework, the preference order over 13 QoS attributes for each opinion sub-group on hotel booking web services will be obtained via RMGDP.

Table 4 show one of the sixty fuzzy QoS dispositions / preference ordering over these attributes from the a certain service consumer. Each consumer of the sixty service consumers can select his / her dispositions in format of (1) on each QoS attribute (a_1, a_2, …a_{13}) based on the available definitions given in the Table 4. He / She also express their preference ordering (in row of o_i) over these attributes which are shown in Table 4. "1" means the most important attribute and "13" represents the least important one. The new fuzzy QoS opinions and feedback as well as their new preference ordering, which will be used to demonstrate FMG-QCMA Moderation Process, follows the same format of the fuzzy QoS opinions defined in Table 4.

Table 4: A Fuzzy QoS Opinions for a certain Service Consumer

wsa^1	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}	a_{11}	a_{12}	a_{13}
o_i	5	10	2	4	9	8	3	1	6	13	7	11	12
x_1	6.0	4.0	5.0	6.0	3.0	5.5	5.5	6.0	5.5	7.0	4.5	2.5	4.0
x_2	6.5	4.5	5.5	6.5	3.5	6.5	6.5	7.0	6.0	7.5	5.5	3.5	4.5
x_3	7.5	5.5	6.5	7.5	4.5	7.5	7.5	8.0	7.0	9.5	6.5	4.5	5.5
x_4	8.0	6.0	7.0	8.0	5.0	8.5	8.5	9.0	7.5	10.0	7.5	5.5	6.0

The similarity threshold, $\tilde{d}_{S_Q}$, is initialized as (0.5, 0.6) and the $f_{c\_S_Q}$ for similarity range is initialized as (0.15, 0.25). If the number of unsatisfactory feedbacks on the recommended web services is more than 3% of the whole opinion population, the resulting cluster is determined as inappropriate. In other words, if the system receives more than 3 unsatisfactory feedbacks from the users, the threshold $\tilde{d}_{S_Q}$ needs to be moderated. Consequently it also re-clusters all fuzzy QoS opinions by going through validation and evaluation process in *Clustering Verification*.

4.2.1 Reaching Consensus: FMGSAM Process

After the required inputs have been obtained, the FMGSAM starts to process the sixty $wsa_{S_Q}^{k}$ in WSA_{S_Q}. One of tasks in Algorithm *Fuzzy_Clustering* is to select an appropriate fuzzy QoS opinion (the first fuzzy opinion which has not been grouped) from the opinion pool to act as a group centre of a specific clustered group, so the other fuzzy QoS opinions (from those which have not been clustered into groups) will be evaluated against the center based on their similarity measurement. The result of the similarity analysis for the first clustered group (G_1) can be shown in Table 5:

Table 5: The Multi-Attributes Similarity Analysis

G_1	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}	a_{11}	a_{12}	a_{13}	sim_result
Sim^{01_01}	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Sim^{01_02}	0.790	0.839	0.706	0.865	0.684	0.938	0.865	0.808	0.659	0.579	0.716	0.519	0.727	0.461
Sim^{01_03}	0.339	0.564	0.733	0.677	0.479	1.000	0.793	0.740	0.659	0.401	0.923	0.692	0.716	0.289
Sim^{01_04}	0.492	0.621	0.733	0.769	0.692	0.667	0.865	0.688	0.808	0.583	0.733	0.586	0.671	0.286
Sim^{01_05}	0.721	0.423	0.667	0.862	0.495	0.564	0.923	0.538	0.769	0.567	0.643	0.755	0.725	0.273
Sim^{01_06}	0.600	0.688	0.706	0.769	0.297	0.865	0.800	0.538	0.725	0.490	0.846	0.544	0.727	0.257
Sim^{01_07}	0.513	0.423	0.533	0.492	0.385	0.867	0.747	0.846	0.790	0.486	0.521	1.000	0.846	0.296
(Sim^{01_08})	0.369	0.604	0.800	1.000	0.489	0.564	1.000	0.346	0.923	0.389	0.586	0.252	0.725	0.184
(Sim^{01_09})	0.790	0.818	0.467	0.361	0.476	0.721	0.800	0.875	0.791	0.437	0.781	0.346	0.587	0.222
~~Sim^{01_10}~~	0.433	0.769	0.857	0.692	0.750	0.692	0.282	0.615	0.929	0.097	0.467	0.519	0.846	~~0.145~~
………	………	………	………	………	………	………	………	………	………	………	………	………	………	………
………	………	………	………	………	………	………	………	………	………	………	………	………	………	………
~~Sim^{10_58}~~	0.481	0.348	0.824	0.538	0.692	0.692	0.149	0.718	0.533	0.692	0.144	0.635	0.923	~~0.023~~
~~Sim^{10_59}~~	0.846	0.242	0.198	0.814	0.431	0.564	0.917	0.641	0.320	0.857	0.333	0.431	0.564	~~0.007~~
~~Sim^{10_60}~~	0.433	0.423	0.875	0.369	0.577	0.867	0.149	0.556	0.852	0.256	0.333	0.564	0.781	~~0.014~~

In Table 5, $Sim_{S_Q}^{jk}$ is represented as Sim^{j_k} which indicates the similarity between $wsa_{S_Q}^{j}$ (group centre) and $wsa_{S_Q}^{k}$. The $Sim_{S_Q}^{jk}$ represented as bold "**Sim^{j_k}**" indicates that $Sim_{S_Q}^{jk}$ is similar to the group *p* and has its full membership to the group. The $Sim_{S_Q}^{jk}$ represented as "(Sim^{j_k})" with regular bracket indicates that $Sim_{S_Q}^{jk}$ only has some degree similarity to the group, so it only has partial membership to the group *p*. Due to the analysis in Table 5 all the sixty fuzzy QoS opinions can be clustered into 13 sub-groups and represented with the index of fuzzy QoS opinion (*k* in $wsa_{S_Q}^{k}$) in Table 6:

Table 6: The Clustered Groups and Opinions

Group_1	1	2	3	4	5	6	7	(8)	(9)	12	(14)	(17)	(19)	22	23	(25)	(28)	(29)	31	(32)	(34)	(38)	42	46	47	(50)	55	(58)	(60)
Group_2	10	(17)	(18)	20	25	(26)	33	35	36	39	(41)	(49)																	
Group_3	11	9	(34)	(40)	(41)	(44)	(48)	50																					
Group_4	13	(8)	(19)	(21)	(26)	(28)	(29)	(30)	34	40	(44)	(56)	58																
Group_5	15	(19)	(21)	(32)	(38)	(43)	45																						
Group_6	16	(14)	(48)	(49)	53																								
Group_7	24	29	(30)	(32)	(38)																								
Group_8	27	(8)	14	57																									
Group_9	37	8	(19)	(21)	(43)	(49)																							
Group_10	51	30																											
Group_11	52	(38)	(44)	48	(60)																								
Group_12	54	43																											
Group_13	59	(21)	49																										

With the 13 clustered sub-groups and each having 13 QoS attributes, there are 169 agreement matrixes (AM) being generated which can be depicted as follow.

$$(AM_{1-a_1})=\begin{bmatrix} 1 & 0.93^{1,2}_{a_1} & 0.88^{1,3}_{a_1} & \cdots & 0.67^{1,58}_{a_1} & 1.00^{1,60}_{a_1} \\ 0.93^{2,1}_{a_1} & 1 & \vdots & \vdots & \vdots & \vdots \\ 0.88^{3,1}_{a_1} & \cdots & 1 & \vdots & \vdots & \vdots \\ \vdots & \cdots & \cdots & 1 & \vdots & \vdots \\ 0.67^{58,1}_{a_1} & \cdots & \cdots & \cdots & 1 & 0.67^{58,60}_{a_1} \\ 1.00^{60,1}_{a_1} & 0.93^{60,2}_{a_1} & \cdots & \cdots & 0.67^{60,58}_{a_1} & 1 \end{bmatrix}_{29\times 29} \quad (AM_{1-a_2})=\begin{bmatrix} 1 & 0.91^{1,2}_{a_2} & 0.73^{1,3}_{a_2} & \cdots & 0.65^{1,58}_{a_2} & 0.69^{1,60}_{a_2} \\ 0.91^{2,1}_{a_2} & 1 & \vdots & \vdots & \vdots & \vdots \\ 0.73^{3,1}_{a_2} & \cdots & 1 & \vdots & \vdots & \vdots \\ \vdots & \cdots & \cdots & 1 & \vdots & \vdots \\ 0.65^{58,1}_{a_2} & \cdots & \cdots & \cdots & 1 & 0.94^{58,60}_{a_2} \\ 0.69^{60,1}_{a_2} & 0.63^{60,2}_{a_2} & \cdots & \cdots & 0.94^{60,58}_{a_2} & 1 \end{bmatrix}_{29\times 29}$$

$$\cdots\cdots \quad (AM_{13-a_{13}})=\begin{bmatrix} 1 & 0.92^{21,49}_{a_{13}} & 0.92^{21,59}_{a_{11}} \\ 0.92^{49,21}_{a_{13}} & 1 & 1.00^{49,59}_{a_{13}} \\ 0.92^{59,21}_{a_{13}} & 1.00^{59,49}_{a_{13}} & 1 \end{bmatrix}_{3\times 3}$$

Figure 14: AMs Generation for All Clustered Groups

After the AMs have been generated, the corresponding AAD, RAD and individual CDC for each fuzzy QoS opinion by each QoS attribute can be derived. Table 7 shows their corresponding results.

Table 7: AAD, RAD and CDC for all Groups

G_1 / a_1	AAD	RAD	β	w_i	CDC
wsa^1	0.8471	0.0358	0.4000	0.0345	0.0352
wsa^2	0.8635	0.0365	0.4000	0.0345	0.0357
wsa^3	0.7794	0.0329	0.4000	0.0345	0.0335
----------------					
wsa^{58}	0.7499	0.0317	0.4000	0.0345	0.0328
wsa^{60}	0.8471	0.0358	0.4000	0.0345	0.0352

G_1 / a_2	AAD	RAD	β	w_i	CDC
wsa^1	0.8079	0.0339	0.4000	0.0345	0.0341
wsa^2	0.7547	0.0317	0.4000	0.0345	0.0328
wsa^3	0.8446	0.0355	0.4000	0.0345	0.0351
----------------					
wsa^{58}	0.7731	0.0325	0.4000	0.0345	0.0333
wsa^{60}	0.8153	0.0342	0.4000	0.0345	0.0343

....

....

G_{13} / a_{12}	AAD	RAD	β	w_i	CDC
wsa^{21}	0.8181	0.3553	0.4000	0.3333	0.3465
wsa^{49}	0.7180	0.3118	0.4000	0.3333	0.3204
wsa^{59}	0.7667	0.3329	0.4000	0.3333	0.3331

G_{13} / a_{13}	AAD	RAD	β	w_i	CDC
wsa^{21}	0.9168	0.3236	0.4000	0.3333	0.3275
wsa^{49}	0.9585	0.3383	0.4000	0.3333	0.3363
wsa^{59}	0.9583	0.3382	0.4000	0.3333	0.3363

In CDC, β is set with 0.4 and each single QoS attribute based fuzzy QoS opinion within the same opinion sub-group is set with the same weight. These parameters setting were determined by experts' opinions according to their experience. With generated CDC of each fuzzy QoS opinions, the group consensus for each opinion sub-group can be obtained and represented as a 13-attributes fuzzy trapezoidal number. Each the sub-group's consensus, which is also represented as fuzzy trapezoidal number, is shown in Table 8.

Table 8: Multi-Groups Consensus by QoS Attributes

G_1 / a_1	CDC	x_1	x_2	x_3	x_4
wsa^1	0.0352	0.2115	0.2291	0.2644	0.2820
wsa^2	0.0357	0.1783	0.2140	0.2496	0.2853
wsa^3	0.0335	0.2347	0.2515	0.2850	0.3018
------------------					
wsa^{58}	0.0328	0.0984	0.1311	0.1639	0.1967
wsa^{60}	0.0352	0.2115	0.2291	0.2644	0.2820
Group Consensus		4.8365	5.7153	6.7153	7.5941

G_1 / a_2	CDC	x_1	x_2	x_3	x_4
wsa^1	0.0341	0.1366	0.1536	0.1878	0.2048
wsa^2	0.0328	0.0984	0.1312	0.1640	0.1968
wsa^3	0.0351	0.2104	0.2279	0.2630	0.2805
------------------					
wsa^{58}	0.0333	0.2329	0.2495	0.2828	0.2994
wsa^{60}	0.0343	0.2060	0.2403	0.2746	0.3089
Group Consensus		4.8948	5.6561	6.6561	7.4174

……………………………………………………………

……………………………………………………………

G_{13} / a_{12}	CDC	x_1	x_2	x_3	x_4
wsa^{21}	0.3465	1.732484	1.905733	2.25223	2.425478
wsa^{49}	0.3204	1.12144	1.281646	1.602057	1.762263
wsa^{59}	0.3331	1.99855	2.165096	2.498188	2.664734
Group Consensus		4.8525	5.3525	6.3525	6.8525

G_{13} / a_{13}	CDC	x_1	x_2	x_3	x_4
wsa^{21}	0.3275	1.309854	1.473585	1.801049	1.964781
wsa^{49}	0.3363	1.345144	1.68143	2.017716	2.354001
wsa^{59}	0.3363	1.345003	1.681253	2.017504	2.353754
Group Consensus		4.0000	4.8363	5.8363	6.6725

4.2.2 Reaching Consensus: RMGDP Process

Based on 13 clustered sub-groups obtained through FMGSAM, there are 13 groups needed to be processed by RMGDP (denoted as $RMGDP_1$, $RMGDP_2$, …, $RMGDP_{12}$, $RMGDP_{13}$) in order to gain their 13 QoS attributes preference orderings. RMGDP starts from transformation phase to generate preference relations for all fuzzy QoS opinions in the corresponding sub-group. Each matrix of preference relations $p^k_{G_p}$ (G_p, P_k in Table 9), which represents all p^k_{ij} defined in RMGDP for $wsa^k_{S_Q}$ in Group p, can be gained via transformation phase in RMGDP shown in Table below:

Table 9: All Matrixes of Preference Relation

G_1, P_1	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}	a_{11}	a_{12}	a_{13}
a_1	0.50	0.71	0.38	0.46	0.67	0.63	0.42	0.33	0.54	0.83	0.58	0.75	0.79
a_2	0.29	0.50	0.17	0.25	0.46	0.42	0.21	0.13	0.33	0.63	0.38	0.54	0.58
a_3	0.63	0.83	0.50	0.58	0.79	0.75	0.54	0.46	0.67	0.96	0.71	0.88	0.92
a_4	0.54	0.75	0.42	0.50	0.71	0.67	0.46	0.38	0.58	0.88	0.63	0.79	0.83
a_5	0.33	0.54	0.21	0.29	0.50	0.46	0.25	0.17	0.38	0.67	0.42	0.58	0.63
a_6	0.38	0.58	0.25	0.33	0.54	0.50	0.29	0.21	0.42	0.71	0.46	0.63	0.67
a_7	0.58	0.79	0.46	0.54	0.75	0.71	0.50	0.42	0.63	0.92	0.67	0.83	0.88
a_8	0.67	0.88	0.54	0.63	0.83	0.79	0.58	0.50	0.71	1.00	0.75	0.92	0.96
a_9	0.46	0.67	0.33	0.42	0.63	0.58	0.38	0.29	0.50	0.79	0.54	0.71	0.75
a_{10}	0.17	0.38	0.04	0.13	0.33	0.29	0.08	0.00	0.21	0.50	0.25	0.42	0.46
a_{11}	0.42	0.63	0.29	0.38	0.58	0.54	0.33	0.25	0.46	0.75	0.50	0.67	0.71
a_{12}	0.25	0.46	0.13	0.21	0.42	0.38	0.17	0.08	0.29	0.58	0.33	0.50	0.54
a_{13}	0.21	0.42	0.08	0.17	0.38	0.33	0.13	0.04	0.25	0.54	0.29	0.46	0.50

■ ■

■ ■

G_{13}, P_{59}	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}	a_{11}	a_{12}	a_{13}
a_1	0.50	0.25	0.58	0.38	0.21	0.33	0.54	0.42	0.63	0.17	0.46	0.13	0.29
a_2	0.75	0.50	0.83	0.63	0.46	0.58	0.79	0.67	0.88	0.42	0.71	0.38	0.54
a_3	0.42	0.17	0.50	0.29	0.13	0.25	0.46	0.33	0.54	0.08	0.38	0.04	0.21
a_4	0.63	0.38	0.71	0.50	0.33	0.46	0.67	0.54	0.75	0.29	0.58	0.25	0.42
a_5	0.79	0.54	0.88	0.67	0.50	0.63	0.83	0.71	0.92	0.46	0.75	0.42	0.58
a_6	0.67	0.42	0.75	0.54	0.38	0.50	0.71	0.58	0.79	0.33	0.63	0.29	0.46
a_7	0.46	0.21	0.54	0.33	0.17	0.29	0.50	0.38	0.58	0.13	0.42	0.08	0.25
a_8	0.58	0.33	0.67	0.46	0.29	0.42	0.63	0.50	0.71	0.25	0.54	0.21	0.38
a_9	0.38	0.13	0.46	0.25	0.08	0.21	0.42	0.29	0.50	0.04	0.33	0.00	0.17
a_{10}	0.83	0.58	0.92	0.71	0.54	0.67	0.88	0.75	0.96	0.50	0.79	0.46	0.63
a_{11}	0.54	0.29	0.63	0.42	0.25	0.38	0.58	0.46	0.67	0.21	0.50	0.17	0.33
a_{12}	0.88	0.63	0.96	0.75	0.58	0.71	0.92	0.79	1.00	0.54	0.83	0.50	0.67
a_{13}	0.71	0.46	0.79	0.58	0.42	0.54	0.75	0.63	0.83	0.38	0.67	0.33	0.50

According to Table 9, the corresponding preference relations aggregation, which is donated as $p^c_{G_p}$, can be obtained via RMGDP with equal weight ($w^i_{G_p}$ = 1 / $|G_p|$). With all generated aggregation of preference relations for each clustered sub-group and equations for *QGNDD* / *QGDD* for each QoS attribute in selected RMGDP$_p$ with the equal weight value w_i for each b_i in *FMQ Moderator* (w_i = 0.083) can be represented as follows:

Table 10: QoS Preference Order Analysis via *QGNDD* / *QGDD*

Group_1	QGNDD	UND Occurs	QGDD
a_1	0.923	$a_2, a_5, a_{10}, a_{11}, a_{12}, a_{13}$	0.506
a_2	0.788	a_{12}, a_{13}	0.406
a_3	0.992	$a_1, a_2, a_4, a_5, a_6, a_9, a_{10}, a_{11}, a_{12}, a_{13}$	0.613
a_4	0.966	$a_1, a_2, a_5, a_6, a_9, a_{10}, a_{11}, a_{12}, a_{13}$	0.557
a_5	0.912	$a_2, a_{10}, a_{11}, a_{12}, a_{13}$	0.495
a_6	0.957	$a_1, a_2, a_5, a_9, a_{10}, a_{11}, a_{12}, a_{13}$	0.543
a_7	1.000	$a_1, a_2, a_3, a_4, a_5, a_6, a_8, a_9, a_{10}, a_{11}, a_{12}, a_{13}$	0.647
a_8	0.998	$a_1, a_2, a_3, a_4, a_5, a_6, a_9, a_{10}, a_{11}, a_{12}, a_{13}$	0.638
a_9	0.956	$a_1, a_2, a_5, a_6, a_{10}, a_{11}, a_{12}, a_{13}$	0.541
a_{10}	0.841	a_2, a_{12}, a_{13}	0.440
a_{11}	0.864	$a_2, a_{10}, a_{12}, a_{13}$	0.457
a_{12}	0.697	a_{13}	0.351
a_{13}	0.633	No UND Occur	0.317

■ ■ ■ ■ ■

Group_13	QGNDD	UND Occurs	QGDD
a_1	0.640	No UND Occur	0.321
a_2	0.977	$a_1, a_3, a_5, a_6, a_7, a_8, a_9, a_{11}, a_{12}, a_{13}$	0.606
a_3	0.747	a_1, a_9	0.380
a_4	1.000	$a_1, a_2, a_3, a_5, a_6, a_7, a_8, a_9, a_{11}, a_{12}, a_{13}$	0.681
a_5	0.924	$a_1, a_3, a_6, a_7, a_9, a_{11}, a_{12}$	0.516
a_6	0.894	$a_1, a_3, a_7, a_9, a_{11}$	0.486
a_7	0.858	a_1, a_3, a_9, a_{11}	0.456
a_8	0.935	$a_1, a_3, a_5, a_6, a_7, a_9, a_{11}, a_{12}$	0.531
a_9	0.726	a_1	0.366
a_{10}	1.000	$a_1, a_3, a_5, a_6, a_7, a_8, a_9, a_{11}, a_{12}, a_{13}$	0.681
a_{11}	0.794	a_1, a_3, a_9	0.411
a_{12}	0.908	$a_1, a_3, a_6, a_7, a_9, a_{11}$	0.501
a_{13}	0.963	$a_1, a_3, a_5, a_6, a_7, a_8, a_9, a_{11}, a_{12}$	0.576

Based on the result in Table 10, the preference ordering over 13 QoS attributes for each sub-group analysed by QGNDD can be represented as below.

$$
\begin{aligned}
o^c_{G_1} &= \{a_7, a_8, a_3, a_4, a_6, a_9, a_1, a_5, a_{11}, a_{10}, a_2, a_{12}, a_{13}\} \\
o^c_{G_2} &= \{a_8 = a_{10}, a_5, a_{11}, a_6 = a_9, a_3, a_4, a_{12}, a_7, a_{13}, a_1, a_2\} \\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\
o^c_{G_{13}} &= \{a_4 = a_{10}, a_2, a_{13}, a_8, a_5, a_{12}, a_6, a_7, a_{11}, a_3, a_9, a_1\}
\end{aligned} \tag{46}
$$

In (46), the preference order over 13 QoS attributes for sub-group G_1, $o^c_{G_1}$, can be explicitly identified by QGNDD. For $o^c_{G_{13}}$ the preference order for a_4 is the same as a_{10}, after they were analyzed by QGNDD. In the case of $o^c_{G_1}$, QGNDD can distinguish most of the attributes by ordering them, but the preference order for a_8 is the same as for a_{10} and a_6 is the same as a_9. Both pair of QoS attributes (a_8, a_{10}) and (a_6, a_9) were further analyzed by QGDD, then $a_{10} > a_8$ and $a_6 = a_9$. Further analysis by QGDD on the preference order for each sub-group can be obtained as follows:

$$\begin{aligned} o^c_{G_1} &= \{a_7, a_8, a_3, a_4, a_6, a_9, a_1, a_5, a_{11}, a_{10}, a_2, a_{12}, a_{13}\} \\ o^c_{G_2} &= \{a_{10}, a_8, a_5, a_{11}, a_6 = a_9, a_3, a_4, a_{12}, a_7, a_{13}, a_1, a_2\} \\ &\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ o^c_{G_{13}} &= \{a_4 = a_{10}, a_2, a_{13}, a_8, a_5, a_{12}, a_6, a_7, a_{11}, a_3, a_9, a_1\} \end{aligned} \quad (47)$$

4.2.3 Marketing Web Service

After the sub-groups have been identified and all the opinions have been allocated into the appropriate groups, it means that the value of CDC for each group is within an acceptable range. In addition, each sub-groups' consensus preference order has been reached. The providers can look up these profiles to advertise their services by registering their services with UDDI. Therefore, the system is ready for recommending the services. Assume Consumer003 in sub-group G_1 requires a suitable hotel booking web service based on his/her disposition on QoS. According to the result of RMGDP analysis for sub-group G_1, the preference order over QoS attributes is: $o^c_{G_1} = \{a_7, a_8, a_3, a_4, a_6, a_9, a_1, a_5, a_{11}, a_{10}, a_2, a_{12}, a_{13}\}$. In other words, the preference order is:

Accuracy > Integrity > Scalability > Capacity > Exception Handling > Accessibility > Performance > Robustness > Interoperability > Availability > Reliability > Security > Friendly GUI (Network Related QoS Requirement).

Figure 15: The Group Preference for Group 1

Assume there are 831 hotels booking web services available, so these are satisfied with functional requirements. These web services will be further analysed by the 13 QoS attributes according to the disposition of sub-group G_1 on each QoS attribute from the most preferable QoS attribute "Accuracy" to the least preferable QoS attribute "Friendly GUI". The inappropriate web services in these 831 will be filtered out according to the order of QoS preference. The following table illustrates the filtering process. In the end of this process, the system only recommends those services that meet the QoS conditions given by Consumer003. In this case, only 7 web services that are satisfied with the consumer's functional and non-functional requirements can be recommended for selection to form a composite service.

Table 11: The Sample Scenario about Recommending Web Services

Preferred QoS attribute	Group Consensus on QoS	Fuzzy Expression	No. of Services via Filtering
Accuracy (a_7)	(5.6, 6.4, 7.4, 8.2)	93% ~ 98%	831 → 470
Integrity (a_8)	(5.4, 6.1, 7.1, 7.9)	Rank (1 ~ 10): 6.1 ~ 7.1	470 → 198
Scalability (a_3)	(5.7, 6.5, 7.5, 8.2)	Rank (1 ~ 10): 6.5 ~ 7.5	198 → 87
Capacity (a_4)	(5.6, 6.2, 7.2, 7.9)	Rank (1 ~ 10): 6.2 ~ 7.2	87 → 38
Exception Handling (a_6)	(5.4, 6.1, 7.1, 7.8)	71% ~ 83%	38 → 24
Accessibility (a_9)	(4.8, 5.5, 6.5, 7.3)	Rank (1 ~ 10): 5.5 ~ 6.5	24 → 19
Performance (a_1)	(4.8, 5.7, 6.7, 7.6)	0.7sec~2.1sec	19 → 14
Robustness (a_5)	(5.3, 6.0, 7.0, 7.8)	Rank (1 ~ 10): 6.0 ~ 7.0	14 → 13
Interoperability (a_{11})	(4.6, 5.4, 6.4, 7.2)	Rank (1 ~ 10): 5.4 ~ 6.4	13 → 12
Availability (a_{10})	(5.0, 5.7, 6.7, 7.5)	Rank (1 ~ 10): 5.7 ~ 6.7	12 → 11
Reliability (a_2)	(4.9, 5.7, 6.7, 7.4)	89% ~ 97%	11 → 10
Security (a_{12})	(3.8, 4.5, 5.5, 6.3)	Transaction Fault Rate 0.089%~0.038%	10 → 8
Friendly GUI (a_{13})	(3.9, 4.6, 5.6, 6.3)	Rank (1 ~ 10): 4.6 ~ 5.6	8 → 7
Conclusion → 7 web services will be recommended			

4.2.4 Process of FMG-QCMA Moderation

The eight new fuzzy QoS opinions $wsa_{S_Q}^{61}$, $wsa_{S_Q}^{62}$, …, $wsa_{S_Q}^{68}$ and three feedback messages with the value *E_Not_Sim* from web service consumers, $wsa_{S_Q}^{25(G_1)}$, $wsa_{S_Q}^{50(G_1)}$ and

$wsa_{S_Q}^{58(G_1)}$, are processed by FMG-QCMA. Through similarity analysis with (8) in operation *Clustering Verification*, the eight new fuzzy QoS opinions were processed in sequence with the thirteen groups (the first fuzzy QoS opinion of each sub-group shown in Table 6, such as $wsa_{S_Q}^{1(G_1)}$, $wsa_{S_Q}^{10(G_2)}$, $wsa_{S_Q}^{11(G_3)}$,.. $wsa_{S_Q}^{59(G_{13})}$) and the sub-group(s) to which each new fuzzy QoS opinion is allocated are illustrated in Table 12:

Table 12: The Re-Clustering with Later Fuzzy QoS Opinions via *Clustering Verification*

		a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}	a_{11}	a_{12}	a_{13}	sim_result	
wsa^{61}	Sim^{01_61}	0.721	0.376	0.429	0.938	0.476	0.577	0.846	0.865	0.396	0.607	0.813	0.593	0.395	0.761	To G_1
wsa^{62}	Sim^{13_62}	0.205	0.510	0.846	0.621	0.813	0.577	0.938	0.426	0.857	0.451	0.462	0.592	0.568	0.618	To G_4
wsa^{63}	Sim^{11_63}	0.462	0.769	0.529	0.375	0.635	0.793	0.923	0.214	0.800	0.636	0.317	0.705	0.846	0.676	To G_3
wsa^{64}	Sim^{13_64}	0.615	0.923	0.574	0.862	0.692	0.326	0.649	0.651	0.308	1.100	0.282	0.456	0.846	0.910	To G_4
wsa^{65}	Sim^{27_65}	0.781	0.857	0.271	0.933	0.646	0.692	0.361	0.800	0.862	0.519	0.544	0.839	0.643	1.286	To G_8
wsa^{66}	Sim^{13_66}	0.527	0.777	0.933	0.554	0.586	0.593	0.869	0.714	0.346	0.538	0.396	0.533	0.497	0.633	To G_4
wsa^{67}	Sim^{11_67}	0.923	0.271	1.000	0.500	0.793	0.577	0.706	0.791	0.692	0.538	0.346	0.543	0.308	0.731	To G_3
wsa^{68}	Sim^{01_68}	0.846	0.582	0.325	0.564	0.692	0.492	0.615	0.889	0.725	0.757	0.568	0.846	0.423	0.912	To G_1

That is, $wsa_{S_Q}^{61}$ and $wsa_{S_Q}^{68}$ are allocated to sub-group G_1; $wsa_{S_Q}^{63}$ and $wsa_{S_Q}^{67}$ become a member of sub-group G_3; $wsa_{S_Q}^{62}$, $wsa_{S_Q}^{64}$ and $wsa_{S_Q}^{66}$ are assigned to sub-group G_4; and $wsa_{S_Q}^{65}$ belongs to sub-group G_8.

For the three feedback messages which are associated with $wsa_{S_Q}^{25(G_1)}$, $wsa_{S_Q}^{50(G_1)}$ and $wsa_{S_Q}^{58(G_1)}$ about inappropriate service recommendation, an event "re-clustering" is triggered to activate the operation *Groups Clustering* because the *m_threshold_distortion* flag is true. As a result, the similarity threshold, $\tilde{d}_{S_Q}$, was moderated from (0.50, 0.60) to (0.52, 0.62), which can be denoted as $\tilde{d}'_{S_Q}$ by the operation *Clustering Verification*. With the moderated similarity threshold $\tilde{d}'_{S_Q}$, all the sixty-eight fuzzy QoS opinions are re-clustered and the new results for AM, AAD, RAD, CDC, Group Consensus and Group Preference order over QoS attributes are obtained through FMGSAM and RMGDP accordingly.

4.3 Review on FMG-QCMA: Precision and Efficiency

In the following experiments, FMG-QCMA can be a marketing web service mechanism based on multi-groups fuzzy QoS disposition consensus of participants. The different weightings over QoS attributes and the relationship among these attributes have been taken into account in order to facilitate the consumers to reach a consensus. The service providers can utilize this result to design and market their services. The approach is a two-layers learning mechanism. In the first layer, the agreement co-efficiency index was used to evaluate the quality of grouping. The initial parameters for arbitrary group boundaries can be adjusted according to the feedback from the group agreement co-efficient. The second learning layer is based on the feedback from the users in order to adjust the number of groups. When the system received too many unsatisfactory recommendations, this implies the grouping is not appropriate and a change of boundaries cannot resolve this issue. So, the number of groups likely needs to increase.

Besides, we attempt to analyze the differences between FMG-QCMA and QCMA in terms of precision in similarity analysis and efficiency in operation. The estimated approaches to generate Agreement Matrix from both methods will be evaluated, as they are the most critical processes in the frameworks. The Agreement Matrix Generation in QCMA QCMA which adopts a single group analysis approach can be expressed as follows.

$$AM_{a_1} = \begin{bmatrix} 0.50_{1,1} & 0.93_{1,2} & 0.88_{1,3} & \cdots & 0.80_{1,11} & \cdots & 1.00_{1,60} \\ 0.93_{2,1} & 0.50_{2,2} & 0.82_{2,3} & \cdots & 0.86_{2,11} & \cdots & 0.93_{2,60} \\ 0.88_{3,1} & 0.82_{3,2} & 0.50_{3,3} & \cdots & 0.71_{3,11} & \cdots & 0.88_{3,60} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0.80_{11,1} & 0.86_{11,2} & 0.71_{11,3} & \cdots & \ddots & \ddots & \vdots \\ \vdots & & & \vdots & \ddots & \ddots & \vdots \\ 1.00_{60,1} & 0.93_{60,2} & 0.88_{60,3} & \cdots & \cdots & \cdots & 0.50_{60,60} \end{bmatrix}_{60\times 60}$$

$$AM_{a_2} = \begin{bmatrix} 0.50_{1,1} & 0.91_{1,2} & 0.73_{1,3} & \cdots & 0.92_{1,11} & \cdots & 0.69_{1,60} \\ 0.91_{2,1} & 0.50_{2,2} & 0.67_{2,3} & \cdots & 0.83_{2,11} & \cdots & 0.63_{2,60} \\ 0.73_{3,1} & 0.67_{3,2} & 0.50_{3,3} & \cdots & 0.80_{3,11} & \cdots & 0.94_{3,60} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0.92_{11,1} & 0.83_{11,2} & 0.80_{11,3} & \cdots & \ddots & \ddots & \vdots \\ \vdots & & & \vdots & \ddots & \ddots & \vdots \\ 0.69_{60,1} & 0.63_{60,2} & 0.94_{60,3} & \cdots & \cdots & \cdots & 0.50_{60,60} \end{bmatrix}_{60\times 60}$$

$$AM_{a_{13}} = \begin{bmatrix} 0.50_{1,1} & 0.73_{1,2} & 0.85_{1,3} & \cdots & 0.73_{1,11} & \cdots & 0.85_{1,60} \\ 0.73_{2,1} & 0.50_{2,2} & 0.62_{2,3} & \cdots & 0.53_{2,11} & \cdots & 0.62_{2,60} \\ 0.85_{3,1} & 0.62_{3,2} & 0.50_{3,3} & \cdots & 0.87_{3,11} & \cdots & 1.00_{3,60} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0.73_{11,1} & 0.53_{11,2} & 0.87_{11,3} & \cdots & \ddots & \ddots & \vdots \\ \vdots & & & \vdots & \ddots & \ddots & \vdots \\ 0.85_{60,1} & 0.62_{60,2} & 1.00_{60,3} & \cdots & \cdots & \cdots & 0.50_{60,60} \end{bmatrix}_{60\times 60}$$

Figure 16: AMs Generation via QCMA

FMG-QCMA adopts multiple sub-groups analysis approach to generate multi-group agreement matrix tables. So, the differences of these two approaches are summarised in Table 13.

Table 13: Similarity Comparison between FMGSAM and SAM

Lowest Similarity	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}	a_{11}	a_{12}	a_{13}
$(AM_1)_{FMGSAM}$	0.47	0.53	0.56	0.53	0.44	0.61	0.77	0.71	0.56	0.42	0.56	0.44	0.47
$(AM)_{SAM}$	0.39	0.47	0.50	0.50	0.44	0.44	0.56	0.50	0.44	0.37	0.50	0.41	0.47
Improvement	21%	12%	11%	6%	0%	38%	38%	41%	29%	14%	13%	6%	0%

....

....

Lowest Similarity	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}	a_{11}	a_{12}	a_{13}
$(AM_{13})_{FMGSAM}$	1.00	0.93	0.64	0.88	0.73	0.69	0.79	0.75	0.69	0.78	0.93	0.67	0.92
$(AM)_{SAM}$	0.39	0.47	0.50	0.50	0.44	0.44	0.56	0.50	0.44	0.37	0.50	0.41	0.47
Improvement	157%	96%	29%	76%	65%	56%	41%	50%	58%	111%	86%	62%	96%

According to Table 13, FMGSAM produces better similarity than the results that QCMA.

Regarding the efficiency, the number of computational operations for generating AM in both SAM and FMGSAM can be summarised as follows:

Table 14: Efficiency Comparison between FMGSAM and SAM

FMGQCMA vs. QCMA	The Calculation of Operational Computation	Total Counts
AM (FMGSAM)	$((29\times 29)_{G_1} + (12\times 12)_{G_2} + (8\times 8)_{G_3} + (13\times 13)_{G_4} + (7\times 7)_{G_5} + (5\times 5)_{G_6} + (5\times 5)_{G_7} + (4\times 4)_{G_8} + (6\times 6)_{G_9} + (2\times 2)_{G_{10}} + (5\times 5)_{G_{11}} + (2\times 2)_{G_{12}} + (3\times 3)_{G_{13}})\times 13$	18,343
AM (SAM)	$(60\times 60)\times 13$	46,800
	Improvement of Operational Computation	60.8%

According to Table 14, FMG-QCMA also has better operation efficiency than QCMA. In this case, it reduces the computational complexity by 60.8%. The effort, however, in forming the clusters is not taken account.

Chapter 5. Conclusion

The study from QCMA to FMG-QCMA focused on the QoS-based web services selection. It performed QoS consensus from unique group structure to multi-groups framework for diversified web service consumers and to alleviate the differences on QoS characteristics in the complicated web services selection.

Regarding the proposed QCMA for unique group structure, it possesses the following features.

1. QCMA is a web service selection mechanism based on fuzzy QoS consensus for a group of participants. The architecture allows them to reach QoS consensus by including a number of activities such as participants' opinion similarity, QoS term preference ordering and QoS fuzzy scale for each QoS term. The contribution of QCMA not only includes the fuzzy inquiry for service selection, but also offers the features to model the QoS preference consensus after aggregating sufficient $wsa_{a_i}^{k}$.
2. QCMA is designed for open and dynamic web environment, such that new opinions and preferences as well as new QoS aspects can be modeled flexibly.

Regarding the proposed FMG-QCMA, elaborating higher precision and efficient QoS-aware selection of web service than unique-group-based scheme (QCMA) and some advantage on marketing web service, FMG-QCMA can further possess the following conclusion.

1. FMG-QCMA is a web service selection mechanism based on fuzzy QoS consensus for multi-groups of participants. The architecture allows them to be fuzzily clustered into appropriate sub-groups to reach QoS consensus by including a number of activities such as participants' opinion similarity and QoS fuzzy scale for each QoS attribute.

2. The similarity analysis for multi-attributes-based QoS defined by W3C [4] can be performed via multi-attributes-based clustering by FMGSAM in FMG-QCMA. The different weight over QoS attributes and similarity for each individual QoS attribute are thought over, too.

3. The FMGSAM achieve higher similarity with multi-groups opinions clustering due to reasoning multi-attributes QoS from different background. The improvement in similarity by FMGSAM than SAM has been proven in experiment.

4. The FMGSAM also achieve higher efficiency under multi-groups framework. The improvement in efficiency can be formulated as $(1 - (O(n_1^2) + O(n_2^2) \ldots + O(n_m^2)) / O(n^2))$, $n = n_1 + n_2 + \ldots. + n_m$.

5. The QoS feedback from web service consumers that closes to "group boundary in similarity" will be clustered as "fuzzily similar" by fuzzy comparison. These QoS that should be also significant in similarity will be thought over so that the consensus based on the similarity analysis will be more credible than hard clustering scheme.

6. With the multi-groups-based framework established by FMGSAM, different preference order generated by RMGDP among different clustered opinion sub-groups based on higher similarity is allowed and generated in higher practicability.

7. The similarity threshold $\tilde{d}_{s_Q}$ can be effectively / efficiently moderated by feedback for delivered fuzzy QoS opinions issued by web service consumers. It makes FMG-QCMA being capable of deciding an appropriate $\tilde{d}_{s_Q}$ to cluster all collected fuzzy QoS opinions according to real perception from web service consumers.

FMG-QCMA also reports its improvements on QCMA in terms of similarity measurement and system efficiency. The FMGSAM achieve higher similarity, as it adopts an effective multi-groups opinions clustering according to service consumers' QoS disposition. It also achieves higher efficiency, as its improvement in efficiency is evident shown in Table 14.

In the dynamic world, customers' perception could be not always kept on fixed level and would be moderated by his / her changeable mind due to growth from learning more experience. In the study of FMG-QCMA we have thought over this factor but still can be further discussed. This dynamic phenomenon could impact the factors to re-cluster fuzzy QoS opinions such as similarity threshold $\tilde{d}_{s_Q}$, weight w_i and corresponded β in CDC generation, etc.. The representation of fuzzy QoS opinions could be also revised to fit in more elaborated customers' perception. These conditions mentioned above would be significant in future work for the series of research in web service selection.

References

[1] Alonso, G., *Web Service – Concepts, Architectures and Applications*, *Springer Verlag*, ISBN 3-540-44008-9, Berlin Heidelberg, 2004

[2] Bellwood, T. et al., UDDI Version 3.0 – UDDI Spec Technical Committee Specification, 19 July 2002, Available at *http://www.uddi.org/pubs/uddi-v3.00-published-20020719.htm*

[3] Benatallah, B., Dumas, M., Sheng, Q.Z., Facilitating the rapid development and scalable orchestration of composite web services, *Distributed and Parallel Databases* 17 (2005), pp. 5–37.

[4] Benatallah, B., Sheng, Q.Z., Dumas, M., The Self-Serv environment for web services composition, *IEEE Internet Computing* 7 (2003) (1), pp. 40–48.

[5] Biron, P. V., Permanente, K., Malhotra, A., XML Scheme Part 2: Datatypes Second Edition, W3C Recommendation 28 October 2004, *http://www.w3.org/TR/xmlschema-2/*

[6] Box, D. et al., Simple Object Access Protocol (SOAP) 1.1, May 2000, Available at *http://www.w3.org/TR/2000/NOTE-SOAP-20000508/*.

[7] Bray, T. et al., Extensible Markup Language (XML) 1.0 (Fifth Edition) W3C Recommendation 26 Nov. 2008, Available at *http://www.w3.org/TR/REC-xml/*.

[8] Chao, K.M. et al., Fuzzy Matchmaking for Web Service, *Proceedings of 19 IEEE Conference on Advanced Network and Information Application, IEEE CS*, 2005, pp 721-726

[9] Chiclana, F., Herrera, F., Herrera-Viedma, E.: A Classification Method of Alternatives for Multiple Preference Ordering Criteria Based on Fuzzy Majority. *Journal of Fuzzy Math*., Vol. 34, 1996, pp 224–229

[10] Chiclana, F., Herrera, F., and Herrera-Viedma, E., Integrating Multiplicative Preference Relations in A Multipurpose Decision-making Model Based on Fuzzy Preference Relations. *Fuzzy Sets and Systems, Elsevier Science*, Vol. 122. , 2001, pp 277–291

[11] Chiclana, F., Herrera, F., Herrera-Viedma, E., Integrating Three Representation Models in Fuzzy Multipurpose Decision Making Based on Fuzzy Preference Relations. *Fuzzy Sets and Systems, Elsevier Science*, Vol. 97., 1998, 33–48

[12] Christensen, E. et al., Web Services Description Language (WSDL) 1.1 W3C Note 15 March 2001, Available at *http://www.w3.org/TR/wsdl.html*.

[13] eMarketer, An Emerging Consensus Paints a Positive Picture, eMarketer 27 November 2006, *http://www.emarketer.com/Article.aspx?R=1004293*

[14] Fazeli, M., Sarbazi-Azad, H., Farivar, R., Parallel Clustering on the Star Graph, *Lecture Notes on Computer Science*, Vol. 3719/2005, pp 287-292

[15] Fung, C.K. et al., A Service-Oriented Composition Framework with QoS Management, *International Journal of Web Services Research*, Vol. 3, Issue 3, 2006, pp 108 - 132

[16] Gunther N. J., *The Practical Performance Analyst*, *McGraw-Hill*, 1998

[17] Hathaway, R., Bezdek, J., Fuzzy c-means clustering of incomplete data, *IEEE Trans. Syst., Man, Cybern.*, vol. 31, no. 5, 2001, pp. 735-744

[18] Herrera-Viedma, E., Herrera, F., Chiclana, F., A Consensus Model for Multiperson Decision Making With Different Preference Structures. *IEEE Transactions on Systems, Man and Cybernetics, IEEE*, Vol.32., 2002, pp 394–402

[19] Herrera, F. et al., A Rational Consensus Model in Group Decision Making Using Linguistic Assessments. *Fuzzy set and Systems, Elsevier Science*, Vol. 88, 1997, pp 31–49

[20] Höppner, F., Klawonn, F., Kruse, R., Fuzzy Cluster Analysis: Methods for Classification, Data Analysis, and Image Recognition, *New York: Wiley*, 1999

[21] Hsu, H.M., Chen, C.T., Aggregation of Fuzzy Opinions under Group Decision Making, *Fuzzy Sets and Systems*, Vol 79, 1996, 279-285

[22] Huang, C.L., Chao, K.M., Lo, C.C., A Moderated Fuzzy Matchmaking for Web Services, *Proceedings of The 5th International Conference on Computer and Information Technology, Shanghai, China, IEEE CS*, 2005, pp 1116-1122.

[23] IEEE, *IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries, New York*, NY, 1990.

[24] Jaeger, M.C., Rojec-Goldmann, G., Mühl, G., Qos aggregation for service composition using workflow patterns. In *Proceedings of the 8th International Enterprise Distributed Object Computing Conference (EDOC 2004), page 149-159, Monterey, California, USA, September 2004. IEEE CS Press.*

[25] Jaeger, M.C., Rojec-Goldmann, G., Mühl. G., Qos aggregation in web service compositions. In: *Proceedings of the 2005 IEEE International Conference on e-Technology, e-Commerce and e-Service (EEE-05), Hong Kong, China, March 2005. IEEE Press*, pp. 181-185.

[26] Jain, A.K., Murty, M.N., Flynn, P.J., Data Clustering: A Review, *ACM Computing Surveys*, Vol 31, issue 3, 1999, pp 264-323

[27] Kannan, R., Vampala, S., Vetta, A., On Clustering: Good, Bad and Spectral, *Journal of the ACM*, Vol. 51, No.3, May 2004, pp 497-515.

[28] Kim, D.Y., Kim, T.Y., Evaluation of Classified Information on Web Agent Using Fuzzy Theory, *International Journal of Fuzzy Logic and Intelligent Systems*, Vol.5, No. 3, (2005.09), pp 216-221,

[29] Liu Y., Ngu, A.H.H., Zeng, L., QoS Computation and Policing in Dynamic Web Service Selection, *ACM WWW Conference*, May 2004, New Yourk, USA, pp. 66-73

[30] Media Mall, US Online Ad Spending Continues Rapid Growth, *Media Mall 27 November 2006, http://www.mediamall.com/blogger/2006_11_01_archive.html*

[31] Menasce, D.A., Composing web services: A QoS view, *IEEE Internet Computing* 8 (2004) (6), pp. 88–90.

[32] Menasce, D.A., QoS issues in Web services, *Internet Computing, IEEE*, Vol. 6, 2002, pp 72-75

[33] Mou, Y. et al., Interactive Web service choice-making based on extended QoS model, *Journal of Zhejiang University – Science A*, 2006, pp 483-492.

[34] Nielsen, H., Leach, P., Lawrence, S., An HTTP Extension Framework, *RFC 2774, Microsoft*, Microsoft, Agranat Systems

[35] Orlovski, S.A., Decision-Making with A Fuzzy Preference Relation. *Fuzzy Sets and Systems, Elsevier Science*, Vol.1, 1978, 155-167

[36] Rajesh S., Arulazi D., Quality of Service for Web Services-Demystification, Limitations, and Best Practices, March 2003.
http://www.developer.com/services/article.php/2027911.

[37] Ran, S., A Model for Web Services Discovery With QoS, *ACM SIGecom Exchanges*, 4(1), March 2003.

[38] Sadeh, L., Fuzzy Set, *Inf. Control*, Vol. 8, 1965, pp. 338 – 353

[39] Sycara, K., Widoff, S., Dynamic Matchmaking Among Heterogeneous Software Agents in Cyberspace, *Journal of AAMAS, 5, Kluwer Academic Publishers*, 2002.

[40] W3C, QoS for Web Service: Requirements and Possible Approaches, *W3C Working Group Note 25 November, http://www.w3c.or.kr/kr-office/TR/2003/ws-qos/*, 2003.

[41] Wang, H. et al., Clustering by Pattern Similarity in Large Data Sets, *Data Mining, Third IEEE International Conference on*, 2003, pp. 187-194

[42] Wang, Y., Stroulia, E., Flexible Interface Matching for Web-Service Discovery, *Proceedings of Fourth International Conference on Web Information System Engineering, IEEE CS*, December, 2003

[43] Wen, J.R., Nie, J.Y., Zhang, H.J., Query Clustering Using User Logs, *ACM Transactions on Information Systems*, Vol. 20, No. 1, January 2002, pp 59-81.

[44] W.M.P. van der Aalst. Don't go with the flow: Web services composition standards exposed. *Jan/Feb 2003 issue of IEEE Intelligent System, pages 72-76, January 2003.*

[45] Xu, R., Survey of Clustering Algorithms, *IEEE Transactions on Neural Networks*, Vol. 16, No. 3, May 2005

[46] Yang, G., Chen, H.P., An Extensible Computing Model for Reputation Evaluation Based on Objective and Automatic Feedbacks, *Advanced Language Processing and Web Information Technology, 2008. ALPIT '08. International Conference on*, 2008, pp 585-592.

Appendix A: Raw Data for QCMA Validation

A.1: The 50 Fuzzy QoS Opinions for QCMA (1/2): QoS Attributes a_1 ~ a_7

a_{ij}: x_j of , according to equation (1) and (2).

	a11	a12	a13	a14	a21	a22	a23	a24	a31	a32	a33	a34	a41	a42	a43	a44	a51	a52	a53	a54	a61	a62	a63	a64	a71	a72	a73	a74
Consumer001	0	0	7	8	0	0	5	9	0	0	4	7	0	0	7	8	0	0	4	8	0	0	5	8	0	0	7	8
Consumer002	0	0	6	8	0	0	6	9	0	0	5	9	0	0	7	9	0	0	3	5	0	0	4	7	0	0	5	9
Consumer003	0	0	8	9	0	0	6	10	0	0	3	8	0	0	5	7	0	0	6	9	0	0	4	8	0	0	7	9
Consumer004	0	0	3	8	0	0	4	7	0	0	6	8	0	0	7	8	0	0	5	7	0	0	6	9	0	0	7	9
Consumer005	0	0	7	9	0	0	6	8	0	0	5	8	0	0	6	8	0	0	5	8	0	0	5	8	0	0	4	8
Consumer006	0	0	5	9	0	0	6	9	0	0	6	9	0	0	7	10	0	0	6	8	0	0	6	9	0	0	6	7
Consumer007	0	0	4	7	0	0	7	9	0	0	7	10	0	0	5	9	0	0	8	10	0	0	4	7	0	0	5	9
Consumer008	0	0	5	7	0	0	5	8	0	0	5	8	0	0	6	9	0	0	8	9	0	0	5	8	0	0	7	8
Consumer009	0	0	5	8	0	0	4	8	0	0	6	8	0	0	7	9	0	0	7	9	0	0	6	9	0	0	6	7
Consumer010	0	0	7	9	0	0	6	9	0	0	4	8	0	0	4	8	0	0	5	7	0	0	5	8	0	0	5	8
Consumer011	0	0	5	7	0	0	5	8	0	0	7	9	0	0	6	9	0	0	7	9	0	0	7	9	0	0	6	9
Consumer012	0	0	6	9	0	0	6	9	0	0	6	9	0	0	5	9	0	0	6	8	0	0	6	8	0	0	5	6
Consumer013	0	0	3	7	0	0	7	10	0	0	5	8	0	0	7	8	0	0	6	9	0	0	5	8	0	0	5	8
Consumer014	0	0	5	8	0	0	3	8	0	0	7	9	0	0	6	9	0	0	5	8	0	0	4	9	0	0	6	8
Consumer015	0	0	6	8	0	0	5	8	0	0	6	7	0	0	7	9	0	0	7	9	0	0	6	7	0	0	7	8
Consumer016	0	0	7	10	0	0	6	9	0	0	6	8	0	0	7	9	0	0	4	7	0	0	5	8	0	0	6	9
Consumer017	0	0	4	7	0	0	4	8	0	0	7	9	0	0	6	9	0	0	6	8	0	0	7	9	0	0	5	8
Consumer018	0	0	8	9	0	0	7	9	0	0	5	7	0	0	5	7	0	0	7	9	0	0	6	8	0	0	6	9
Consumer019	0	0	7	9	0	0	6	9	0	0	6	8	0	0	7	9	0	0	8	10	0	0	6	9	0	0	7	8
Consumer020	0	0	5	8	0	0	5	8	0	0	6	8	0	0	8	10	0	0	5	8	0	0	5	8	0	0	6	9
Consumer021	0	0	6	9	0	0	6	8	0	0	7	8	0	0	4	7	0	0	6	8	0	0	6	7	0	0	6	7
Consumer022	0	0	4	8	0	0	7	9	0	0	5	8	0	0	7	9	0	0	4	9	0	0	6	8	0	0	5	8
Consumer023	0	0	7	8	0	0	6	8	0	0	6	7	0	0	6	7	0	0	7	8	0	0	5	8	0	0	5	8
Consumer024	0	0	5	8	0	0	5	9	0	0	4	6	0	0	5	8	0	0	4	9	0	0	5	9	0	0	5	9
Consumer025	0	0	6	8	0	0	7	8	0	0	4	7	0	0	7	9	0	0	6	7	0	0	6	8	0	0	7	8
Consumer026	0	0	6	9	0	0	6	9	0	0	7	10	0	0	7	8	0	0	5	8	0	0	6	7	0	0	6	8
Consumer027	0	0	6	7	0	0	4	7	0	0	7	8	0	0	6	8	0	0	3	8	0	0	4	9	0	0	7	9
Consumer028	0	0	5	9	0	0	7	9	0	0	5	8	0	0	5	8	0	0	7	9	0	0	7	8	0	0	6	9
Consumer029	0	0	4	8	0	0	8	10	0	0	6	7	0	0	8	9	0	0	7	9	0	0	6	10	0	0	7	8
Consumer030	0	0	6	9	0	0	5	8	0	0	3	7	0	0	7	9	0	0	5	7	0	0	5	7	0	0	5	9
Consumer031	0	0	7	9	0	0	6	9	0	0	7	9	0	0	8	10	0	0	6	8	0	0	6	9	0	0	6	8
Consumer032	0	0	5	8	0	0	4	7	0	0	6	9	0	0	7	8	0	0	4	8	0	0	5	6	0	0	7	9
Consumer033	0	0	8	10	0	0	5	8	0	0	5	8	0	0	5	9	0	0	7	9	0	0	4	7	0	0	6	9
Consumer034	0	0	5	9	0	0	6	9	0	0	7	9	0	0	7	10	0	0	6	8	0	0	7	9	0	0	7	8
Consumer035	0	0	7	8	0	0	7	8	0	0	6	8	0	0	7	9	0	0	5	8	0	0	6	8	0	0	8	9
Consumer036	0	0	6	9	0	0	5	7	0	0	5	7	0	0	5	7	0	0	4	7	0	0	5	7	0	0	6	7
Consumer037	0	0	7	9	0	0	7	9	0	0	6	8	0	0	6	9	0	0	6	8	0	0	7	8	0	0	4	9
Consumer038	0	0	5	8	0	0	8	9	0	0	4	8	0	0	4	7	0	0	5	9	0	0	6	8	0	0	5	8
Consumer039	0	0	7	9	0	0	4	8	0	0	6	9	0	0	7	8	0	0	7	8	0	0	5	9	0	0	6	8
Consumer040	0	0	4	7	0	0	5	7	0	0	7	8	0	0	5	7	0	0	5	7	0	0	6	8	0	0	5	7
Consumer041	0	0	6	8	0	0	6	9	0	0	5	8	0	0	6	9	0	0	6	8	0	0	7	9	0	0	5	6
Consumer042	0	0	5	8	0	0	5	7	0	0	6	7	0	0	7	8	0	0	6	9	0	0	6	7	0	0	4	8
Consumer043	0	0	7	9	0	0	6	8	0	0	7	8	0	0	5	7	0	0	5	8	0	0	6	8	0	0	6	9
Consumer044	0	0	6	9	0	0	7	8	0	0	6	9	0	0	6	8	0	0	7	8	0	0	7	9	0	0	6	8
Consumer045	0	0	6	10	0	0	5	8	0	0	7	8	0	0	7	8	0	0	4	6	0	0	5	7	0	0	5	7
Consumer046	0	0	7	7	0	0	6	6	0	0	5	7	0	0	5	6	0	0	7	9	0	0	6	7	0	0	7	8
Consumer047	0	0	5	6	0	0	8	8	0	0	6	8	0	0	6	8	0	0	6	9	0	0	5	6	0	0	5	8
Consumer048	0	0	4	8	0	0	7	9	0	0	7	8	0	0	7	8	0	0	5	8	0	0	7	8	0	0	6	7
Consumer049	0	0	7	9	0	0	5	8	0	0	6	8	0	0	6	9	0	0	7	8	0	0	4	9	0	0	5	9
Consumer050	0	0	8	9	0	0	6	9	0	0	7	9	0	0	5	8	0	0	6	9	0	0	6	8	0	0	6	8

A.2: The 50 Fuzzy QoS Opinions for QCMA (2/2): QoS Attributes a_8 ~ a_{13}

a_{ij}: x_j of , according to equation (1) and (2).

	a81	a82	a83	a84	a91	a92	a93	a94	aa1	aa2	aa3	aa4	ab1	ab2	ab3	ab4	ac1	ac2	ac3	ac4	ad1	ad2	ad3	ad4
Consumer001	0	0	3	8	0	0	6	9	0	0	9	10	0	0	6	7	0	0	4	8	0	0	6	9
Consumer002	0	0	6	8	0	0	5	7	0	0	6	9	0	0	7	10	0	0	7	9	0	0	4	7
Consumer003	0	0	5	8	0	0	5	7	0	0	6	9	0	0	3	7	0	0	5	7	0	0	6	9
Consumer004	0	0	4	7	0	0	7	10	0	0	5	7	0	0	3	8	0	0	6	9	0	0	4	7
Consumer005	0	0	7	9	0	0	6	8	0	0	6	8	0	0	6	8	0	0	5	6	0	0	6	8
Consumer006	0	0	6	8	0	0	5	7	0	0	6	9	0	0	6	7	0	0	6	8	0	0	4	9
Consumer007	0	0	7	9	0	0	4	8	0	0	5	7	0	0	5	8	0	0	4	7	0	0	5	7
Consumer008	0	0	5	7	0	0	6	8	0	0	6	9	0	0	4	8	0	0	5	6	0	0	6	8
Consumer009	0	0	6	8	0	0	5	7	0	0	5	7	0	0	5	6	0	0	5	7	0	0	4	7
Consumer010	0	0	8	10	0	0	6	9	0	0	5	7	0	0	4	8	0	0	4	8	0	0	5	8
Consumer011	0	0	6	8	0	0	5	8	0	0	3	8	0	0	7	9	0	0	5	7	0	0	3	8
Consumer012	0	0	7	7	0	0	5	7	0	0	6	7	0	0	6	8	0	0	3	6	0	0	5	8
Consumer013	0	0	5	8	0	0	6	8	0	0	5	8	0	0	5	7	0	0	6	7	0	0	6	7
Consumer014	0	0	6	9	0	0	4	8	0	0	6	8	0	0	4	7	0	0	5	8	0	0	4	7
Consumer015	0	0	4	7	0	0	5	8	0	0	7	9	0	0	5	8	0	0	4	7	0	0	5	7
Consumer016	0	0	4	7	0	0	6	7	0	0	5	8	0	0	7	9	0	0	3	6	0	0	5	8
Consumer017	0	0	5	7	0	0	5	8	0	0	6	8	0	0	5	6	0	0	4	7	0	0	5	6
Consumer018	0	0	3	8	0	0	7	9	0	0	6	7	0	0	6	7	0	0	4	8	0	0	6	7
Consumer019	0	0	7	8	0	0	6	8	0	0	7	8	0	0	5	9	0	0	5	7	0	0	5	8
Consumer020	0	0	6	7	0	0	5	8	0	0	7	8	0	0	5	8	0	0	4	6	0	0	4	5
Consumer021	0	0	7	9	0	0	4	7	0	0	8	10	0	0	6	7	0	0	6	7	0	0	5	6
Consumer022	0	0	6	8	0	0	5	8	0	0	6	8	0	0	4	6	0	0	4	8	0	0	4	8
Consumer023	0	0	5	7	0	0	6	6	0	0	5	9	0	0	7	9	0	0	5	7	0	0	6	6
Consumer024	0	0	4	8	0	0	3	8	0	0	7	8	0	0	5	8	0	0	3	6	0	0	5	8
Consumer025	0	0	6	9	0	0	6	8	0	0	4	7	0	0	4	9	0	0	4	7	0	0	4	5
Consumer026	0	0	5	7	0	0	5	8	0	0	6	8	0	0	8	10	0	0	5	8	0	0	5	7
Consumer027	0	0	7	8	0	0	4	7	0	0	5	7	0	0	5	9	0	0	4	6	0	0	4	7
Consumer028	0	0	6	9	0	0	5	7	0	0	6	9	0	0	6	7	0	0	6	7	0	0	3	6
Consumer029	0	0	5	8	0	0	6	8	0	0	7	8	0	0	6	7	0	0	5	7	0	0	5	8
Consumer030	0	0	6	7	0	0	4	8	0	0	5	6	0	0	5	8	0	0	5	6	0	0	6	7
Consumer031	0	0	4	8	0	0	5	7	0	0	6	7	0	0	4	7	0	0	4	7	0	0	5	8
Consumer032	0	0	6	9	0	0	6	8	0	0	7	8	0	0	5	8	0	0	3	8	0	0	4	6
Consumer033	0	0	5	7	0	0	4	7	0	0	5	8	0	0	6	7	0	0	4	6	0	0	5	7
Consumer034	0	0	4	6	0	0	5	7	0	0	7	9	0	0	6	9	0	0	5	7	0	0	6	8
Consumer035	0	0	7	8	0	0	4	8	0	0	5	8	0	0	4	6	0	0	5	8	0	0	5	7
Consumer036	0	0	6	8	0	0	5	8	0	0	6	8	0	0	5	7	0	0	6	7	0	0	4	8
Consumer037	0	0	6	9	0	0	6	7	0	0	7	8	0	0	5	9	0	0	6	9	0	0	5	7
Consumer038	0	0	6	8	0	0	4	7	0	0	6	7	0	0	7	8	0	0	4	6	0	0	6	8
Consumer039	0	0	5	9	0	0	7	8	0	0	6	7	0	0	6	7	0	0	5	7	0	0	5	6
Consumer040	0	0	6	8	0	0	5	7	0	0	4	8	0	0	5	8	0	0	6	8	0	0	4	9
Consumer041	0	0	5	9	0	0	6	8	0	0	7	8	0	0	6	9	0	0	5	6	0	0	6	8
Consumer042	0	0	4	7	0	0	5	8	0	0	6	7	0	0	5	7	0	0	4	7	0	0	5	8
Consumer043	0	0	6	8	0	0	5	8	0	0	5	6	0	0	6	8	0	0	5	8	0	0	5	7
Consumer044	0	0	5	9	0	0	6	8	0	0	4	7	0	0	5	9	0	0	6	9	0	0	4	8
Consumer045	0	0	6	7	0	0	6	8	0	0	5	8	0	0	6	8	0	0	5	7	0	0	5	7
Consumer046	0	0	7	8	0	0	7	7	0	0	6	7	0	0	5	8	0	0	4	8	0	0	6	8
Consumer047	0	0	5	7	0	0	6	8	0	0	6	7	0	0	6	9	0	0	5	8	0	0	5	5
Consumer048	0	0	6	6	0	0	5	6	0	0	4	7	0	0	6	9	0	0	6	9	0	0	4	6
Consumer049	0	0	5	7	0	0	6	7	0	0	3	7	0	0	6	8	0	0	4	7	0	0	5	7
Consumer050	0	0	7	8	0	0	5	8	0	0	7	8	0	0	5	9	0	0	5	9	0	0	6	8

A.3: The Order Preference for 50 Participants in QCMA

	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}	a_{11}	a_{12}	a_{13}
Consumer001	5	10	2	4	9	8	3	1	6	13	7	11	12
Consumer002	7	11	1	3	6	8	4	2	9	13	5	10	12
Consumer003	13	7	4	2	5	8	1	3	9	11	6	12	10
Consumer004	6	8	4	1	10	11	2	3	7	12	5	9	13
Consumer005	8	4	1	3	6	5	2	7	9	10	11	12	13
Consumer006	9	10	3	1	2	7	4	6	8	11	5	13	12
Consumer007	2	5	7	9	6	8	1	3	4	10	12	11	13
Consumer008	12	7	1	4	10	11	3	8	5	6	9	2	13
Consumer009	3	10	8	12	11	5	2	1	7	9	6	4	13
Consumer010	12	13	3	8	9	4	11	5	6	2	1	7	10
Consumer011	9	11	8	12	13	7	6	3	1	10	5	4	2
Consumer012	5	8	1	9	3	2	4	7	10	11	6	12	13
Consumer013	12	2	8	5	13	9	1	4	6	7	3	11	10
Consumer014	1	6	2	5	4	8	10	7	3	9	12	11	13
Consumer015	2	10	9	8	6	5	1	11	12	4	7	13	3
Consumer016	7	4	5	1	2	6	9	8	10	13	3	11	12
Consumer017	10	12	8	11	4	3	2	1	5	6	7	9	13
Consumer018	9	11	4	5	10	12	8	3	7	2	6	1	13
Consumer019	11	8	9	3	5	4	1	2	12	10	7	13	6
Consumer020	11	12	8	10	3	9	7	5	4	6	1	2	13
Consumer021	10	3	11	1	13	4	2	8	7	5	9	12	6
Consumer022	4	10	8	5	7	6	1	2	3	9	11	12	13
Consumer023	12	9	1	4	2	3	5	7	6	8	10	11	13
Consumer024	5	1	6	8	9	2	4	3	11	7	12	13	10
Consumer025	8	13	3	1	5	6	12	9	4	2	7	10	11
Consumer026	12	13	2	9	3	4	1	5	7	6	8	11	10
Consumer027	2	6	1	12	5	13	11	8	4	10	7	3	9
Consumer028	8	4	3	2	13	6	9	1	7	5	11	12	10
Consumer029	11	9	8	6	1	2	4	3	5	7	10	12	13
Consumer030	3	2	6	5	11	8	7	13	4	1	12	10	9
Consumer031	9	12	8	3	11	6	4	1	2	5	7	10	13
Consumer032	1	13	9	11	10	2	5	3	8	6	4	12	7
Consumer033	11	13	8	1	12	3	6	9	10	5	2	7	4
Consumer034	8	9	5	11	13	10	1	2	6	3	4	12	7
Consumer035	12	13	6	8	1	2	4	5	7	9	3	10	11
Consumer036	9	13	8	12	5	6	11	4	1	2	7	3	10
Consumer037	12	13	6	4	11	8	1	7	9	5	10	2	3
Consumer038	2	11	9	5	7	1	4	6	10	8	3	12	13
Consumer039	12	13	8	2	4	3	6	5	1	7	9	11	10
Consumer040	8	2	9	12	13	5	1	3	6	4	7	10	11
Consumer041	5	12	9	11	1	8	13	2	10	6	3	7	4
Consumer042	8	13	1	5	2	11	4	6	3	9	12	10	7
Consumer043	3	7	8	1	9	5	2	11	12	13	10	6	4
Consumer044	7	4	3	9	13	1	8	5	2	10	6	11	12
Consumer045	10	11	7	4	5	3	1	9	8	2	12	13	6
Consumer046	1	9	6	10	12	4	8	2	3	5	7	11	13
Consumer047	8	13	1	11	7	6	3	4	2	9	10	5	12
Consumer048	6	4	8	11	12	1	7	9	5	13	3	2	10
Consumer049	13	7	6	1	4	12	11	3	10	2	9	8	5
Consumer050	9	11	8	10	7	5	6	1	4	13	3	12	2

Appendix B: Algorithm Fuzzy Clustering

Algorithm **Fuzzy_Clustering(**WSA_{S_Q}**)**

/* The algorithm assumes the definitions: K, the set of consumers; $wsa^k_{S_Q}$, the set of trapezoidal opinions for consumer, *k*, over the set of attributes, S_Q; WSA_{S_Q} is the collection of all the $wsa^k_{S_Q}$; and G_p is a subset of K containing the consumers in cluster, p.

1. $WSA\_temp_{S_Q} \leftarrow WSA_{S_Q}$; /* Copy all incoming opinions into a temporary set for clustering.
2. $p \leftarrow 0$; /* *p* is set as subgroup ID and initialized as 0
3. *while* $WSA\_temp_{S_Q}$ *is not empty* /* Clustering Loop for a created group.
4. $j \leftarrow \min\{k \mid k \in K, wsa^k_{S_Q} \in WSA\_temp_{S_Q}$ and $wsa^k_{S_Q} \notin G_{p'}.Abs\_Sim, G_{p'} \in all\ created\ G_p\}$.
5. $p \leftarrow p + 1$; /* Set Subgroup ID.
6. $max\_p \leftarrow p$; /* Record the maximum group index in the clustering.
7. $wsa^{G_p}_{S_Q} \leftarrow wsa^j_{S_Q}$; /* Set group centre for G_p with the minimum index of opinion from step 4.
8. $WSA\_temp_{S_Q} \leftarrow WSA\_temp_{S_Q} - \{wsa^j_{S_Q}\}$; /* Remove the opinion $wsa^j_{S_Q}$ from evaluated list.
9. $cluster\_temp_{S_Q} \leftarrow WSA\_temp_{S_Q}$; /* Copy the temporary set to the other set for comparison in clustering.
10. $G_p.Abs\_Sim \leftarrow \{wsa^{G_p}_{S_Q}\}$; /* Insert group centre to "Similar Area" in G_p.
11. $n^{G_p}_t \leftarrow 1$; /* Initialize $n^{G_p}_t$: no. of $wsa^j_{S_Q}$ in G_p.
12. *while* $cluster\_temp_{S_Q}$ *is not empty* /* Cluster all evaluated opinions in set for

comparison.

13. $j \leftarrow \min \{k| k \in K, wsa_{S_Q}^{k} \in cluster\_temp_{S_Q}\}$;

14. *select* $wsa_{S_Q}^{j}$ in $cluster\_temp_{S_Q}$;

15. *if* $SimVerify(Sim_{S_Q}^{G_p j}$, "$\geqq$", $\tilde{d}_{S_Q}) > 0$ *then* /* $Sim_{S_Q}^{G_p j} \geqq \tilde{d}_{S_Q}$.

16. $n_t^{G_p} \leftarrow n_t^{G_p} + 1$;

17. *if* $SimVerify(Sim_{S_Q}^{G_p j}$, "$\gtrsim$", $\tilde{d}_{S_Q}) > 0$ *then* /* $Sim_{S_Q}^{G_p j} \gtrsim \tilde{d}_{S_Q}$, $wsa_{S_Q}^{j}$ should be clustered.

18. $G_p.Abs\_Sim \leftarrow G_p.Abs\_Sim + \{wsa_{S_Q}^{j}\}$;/* Insert evaluated opinion into "Similar Area" in G_p.

19. $WSA\_temp_{S_Q} \leftarrow WSA\_temp_{S_Q} - \{wsa_{S_Q}^{j}\}$; /* Remove the evaluated opinion due to step 17.

20. *else*

21. $G_p.Fuz\_Sim \leftarrow G_p.Fuz\_Sim + \{wsa_{S_Q}^{j}\}$; /* Insert evaluated opinion into "like Similar Area" but the evaluated opinion will be kept for next round.

22. *endif* /* *if* $(Sim_{S_Q}^{G_p j} \gtrsim \tilde{d}_{S_Q})$.

23. *endif* /* *if* $(Sim_{S_Q}^{G_p j} \geqq \tilde{d}_{S_Q})$.

24. $cluster\_temp_{S_Q} \leftarrow cluster\_temp_{S_Q} - \{wsa_{S_Q}^{j}\}$; /*Remove $wsa_{S_Q}^{j}$ from the evaluation for comparison.

25. *end while* $cluster\_temp_{S_Q}$ *is not empty* /* Go evaluation for next opinion.

26. *end while* $WSA\_temp_{S_Q}$ *is not empty* /* Go to next clustered group.

27. *end Algorithm* Fuzzy_Clustering(WSA_{S_Q});

Appendix C: Algorithm *SimVerifier*

***Algorithm* SimVerifier(** $Sim_{S_Q}^{jk}$, *sim_operator*, $\tilde{d}_{S_Q}$ **)**

/* *sim_result*: an indicator for the similarity verification by comparison between $Sim_{S_Q}^{jk}$ and $\tilde{d}_{S_Q}$.

1. *sim_result* ← 0; /* Initialize *sim_result* as 0.

/ * Do similarity comparison over 13 QoS attributes and convert to *sim_result* for further analysis.

2. *for i* = 1 to 13

3. *if* $(so_{a_i}^{jk} \times Sim_{a_i}^{jk}) > d_{S_Q}^{u}$ *then* *sim_result* ← *sim_result* + $\left| so_{a_i}^{jk} \times Sim_{a_i}^{jk} - d_{S_Q}^{u} \right|$;

4. *if* $(so_{a_i}^{jk} \times Sim_{a_i}^{jk}) < d_{S_Q}^{l}$ *then* *sim_result* ← *sim_result* - $\left| so_{a_i}^{jk} \times Sim_{a_i}^{jk} - d_{S_Q}^{l} \right|$;

5. *end for i* = 1 to 13

/* Aug is a variable to augment a value to become distinguishable. In this case 3 is sufficient.

6. *Aug* = 3; *sim_result* ← *Aug* ×(*sim_result* / 13);

7. *Case sim_operator of*

8. "≧": /* ($Sim_{S_Q}^{jk}$ ≧ $\tilde{d}_{S_Q}$) is recognized.

9. *if* ($f_{c\_S_Q}^{l}$ ≦ *sim_result* ≦ 1) *then* *return* (*sim_result*) *else* *return* (-1);

10. "⋝": /* ($Sim_{S_Q}^{jk}$ ⋝ $\tilde{d}_{S_Q}$) is recognized.

11. *if* ($f_{c\_S_Q}^{u}$ ≦ *sim_result* ≦ 1) *then* *return* (*sim_result*) *else* *return* (-1);

12. "≅": /* ($Sim_{S_Q}^{jk}$ ≅ $\tilde{d}_{S_Q}$) is recognized.

13. *if* ($f_{c\_S_Q}^{l}$ ≦ *sim_result* < $f_{c\_S_Q}^{u}$) *then return* (*sim_result*) *else* *return* (-1);

14. "≨": /* ($Sim_{S_Q}^{jk}$ ≨ $\tilde{d}_{S_Q}$) is recognized.

15. *if* (0 ≦ *sim_result* < $f_{c\_S_Q}^{u}$) *then* *return* (*sim_result*) *else* *return* (-1);

16. "≾": /* ($Sim_{S_Q}^{jk}$ ≾ $\tilde{d}_{S_Q}$) is recognized.

if (0 ≦ *sim_result* < $f^{l}_{c\_S_Q}$) *then* *return* (*sim_result*) *else* *return* (-1);

end Case; /* *sim_operator*

End Algo. SimVerifier($Sim^{jk}_{S_Q}$, *sim_operator*, $\tilde{d}_{S_Q}$));

Appendix D: Algorithm *Clustering Verification*

Algorithm **Clustering_Verification(** $wsa_{S_Q}^{j}$, *s_feedback*, *group_ID*)

/* Identify if $wsa_{S_Q}^{j}$ was on "Similar Area" or "Like Similar Area", from the first group it was allocated.

p_Sim_Type ← GetSimType($wsa_{S_Q}^{j}$, *group_ID*); /* Return if $wsa_{S_Q}^{j}$ is *E_Fail_CDC, E_Fuz_Sim or E_Abs_Sim.*
if Validation(*group_ID*) is true *then* /* *group_ID* is valid.
    /* Verify the cases of *s_feedback*: Fail CDC (detecting by CDC threshold) or later mismatched feedback.
    Case s_feedback of
    /* Verify the conditions if the CDC for $wsa_{S_Q}^{j}$ is less than the CDC threshold of evaluated clustered group.
    E_Fail_CDC:
        m_count_fdistance_too_long ← *m_count_fdistance_too_long* + 1;
        if m_count_fdistance_too_long ≧ *m_threshold_distortion then*
            if $d_{S_Q}^{l} \geqq 0.02$ /* Moderate $d_{S_Q}^{l}$.
                $d_{S_Q}^{l} \leftarrow d_{S_Q}^{l} - 0.02$;
                $d_{S_Q}^{u} \leftarrow d_{S_Q}^{u} - 0.02$;
                Fuzzy_Clustering(WSA_{S_Q} **);**
            endif /* $d_{S_Q}^{l} \geqq 0.02$.
        endif / if m_count_fdistance_too_long* ≧ *m_threshold_distortion.*
    /* Verify the conditions if $wsa_{S_Q}^{j}$ was allocated into mismatched area..
    Otherwise:
        Case p_Sim_Type of
        E_Fuz_Sim:
            if s_feedback = E_Not_Sim then

m_count_fdistance_too_long ← *m_count_fdistance_too_long* + 1;

17. *if s_feedback = E_Abs_Sim then*
  m_count_fdistance_too_short ← *m_count_fdistance_too_short* + 1;

18. *E_Abs_Sim:*

19. *if* (*s_feedback = E_Not_Sim*) or (*s_feedback = E_Fuz_Sim*) *then*

20. *m_count_fdistance_too_long* ← *m_count_fdistance_too_long* + 1;

21. *endif* /* *if* (*s_feedback = E_Not_Sim*) or (*s_feedback = E_Fuz_Sim*)

22. *Otherwise:* /* Allocate this $wsa_{S_Q}^{j}$ into appropriate group.

23. *for p = 1 to max_p*

24. *if* $SimVerify(Sim_{S_Q}^{G_p j}$, "≧", $\tilde{d}_{S_Q}) > 0$ *then* /* $Sim_{S_Q}^{G_p j} \geqq \tilde{d}_{S_Q}$.

25. $n_t^{G_p} \leftarrow n_t^{G_p} + 1$;

26. *if* $SimVerify(Sim_{S_Q}^{G_p j}$, "≳", $\tilde{d}_{S_Q}) > 0$ *then* /* $Sim_{S_Q}^{G_p j} \gtrsim \tilde{d}_{S_Q}$, $wsa_{S_Q}^{j}$ should be clustered.

27. $G_p.Abs\_Sim \leftarrow G_p.Abs\_Sim + \{wsa_{S_Q}^{j}\}$; /* Insert opinion into "Similar Area" in G_p.

28. $WSA\_temp_{S_Q} \leftarrow WSA\_temp_{S_Q} - \{wsa_{S_Q}^{j}\}$; /* Remove the opinion due to step 17.

29. *break;* /* Terminate Algorithm Clustering_Verification when just allocate $wsa_{S_Q}^{j}$.

30. *else*
  /* Insert opinion into "like Similar Area" but the evaluated opinion will be kept for next round.

31. $G_p.Fuz\_Sim \leftarrow G_p.Fuz\_Sim + \{wsa_{S_Q}^{j}\}$;

32. *endif* /* *if* $(Sim_{S_Q}^{G_p j} \gtrsim \tilde{d}_{S_Q})$

33. *endif* /* *if* $(Sim_{S_Q}^{G_p j} \geqq \tilde{d}_{S_Q})$

34. *end for p = 1 to max_p*

35. *end Case; /* p_Sim_Type*

/* Determine if re-clustering by moderated threshold for similarity should be enabled or not.

if m_count_fdistance_too_long ≧ *m_threshold_distortion then*

if $d^{l}_{S_Q} \geqq 0.02$ /* Moderate $d^{l}_{S_Q}$.

$d^{l}_{S_Q} \leftarrow d^{l}_{S_Q} - 0.02$;

$d^{u}_{S_Q} \leftarrow d^{u}_{S_Q} - 0.02$;

Fuzzy_Clustering(WSA_{S_Q});

endif /* $d^{l}_{S_Q} \geqq 0.02$

endif / if m_count_fdistance_too_long* ≧ *m_threshold_distortion.*

if m_count_fdistance_too_short ≧ *m_threshold_distortion then*

if $d^{u}_{S_Q} \leqq 0.98$ /* Moderate $d^{u}_{S_Q}$.

$d^{u}_{S_Q} \leftarrow d^{u}_{S_Q} + 0.02$;

$d^{l}_{S_Q} \leftarrow d^{l}_{S_Q} + 0.02$;

Fuzzy_Clustering(WSA_{S_Q});

endif /* $d^{u}_{S_Q} \leqq 0.98$.

endif / if m_count_fdistance_too_short* ≧ *m_threshold_distortion.*

end Case; / s_feedback.*

endif / if Validation*(*group_ID*) is true.

end Algorithm Clustering_Verification($wsa^{j}_{S_Q}$, *s_feedback*, *group_ID*);

個人簡歷

博士論文作者: 林謂立

1968 年出生於台灣省宜蘭縣。

1986 年畢業於宜蘭高級中學。

1990 年畢業於私立大同工學院資訊工程學系，獲得工程學士學位。

2003 年畢業於國立交通大學管理學院在職專班資管組，獲得管理學碩士學位。

2003 年~2009 年進入國立交通大學資訊管理研究所博士班，進修資訊管理學博士學位。

在進修博士學位期間，曾發表六篇論文，其中包含兩篇期刊論文於 A 級 SCI 期刊(JCSS, Journal of Computer and System Sciences)及四篇學術會議論文(其中錄取於 AINA2006 之學術會議論文集為 EI 學術會議論文)。發表之論文列舉如下:

Journal Paper (期刊論文):

1. Wei-Li Lin, Chi-Chun Lo, Kuo-Ming Chao, Muhammad Younas, Consumer-centric QoS-aware selection of web services, ***Journal of Computer and System Sciences***, pp 211-231, 2008 (**SCI, 2007 Impact Factor: 1.185**, accepted on 31 Oct 2006, available online on 24 April 2007)

2. Wei-Li Lin, Chi-Chun Lo, Kuo-Ming Chao, Nick Godwin, Web Services for Multi-Group QoS Consensus, ***Journal of Computer and System Sciences***, (**SCI, 2007 Impact Factor: 1.185**, accepted on 16 June 2009)

Conference Paper (會議論文):

1. Wei-Li Lin, Chi-Chun Lo, Kuo-Ming Chao, Nick Godwin, Fuzzy Similarity Clustering for Intelligent Consumer-centric QoS-aware Selection of Web Services, The Second International Workshop on Adaptive Systems in Heterogeneous Environments (ASHEs 2009), Fukuoka, Japan, 16 Mar 2009.
2. Wei-Li Lin, Chi-Chun Lo, Kuo-Ming Chao, Muhammad Younas, Fuzzy Consensus on QoS in Web Services Discovery, IEEE AINA 2006 conference, Vienna, Austria, Proceeding Vol.1 791-798, 18 Apr 2006 **(EI)**.
3. Wei-Li Lin, Chi-Chun Lo, Jay Wu, An XCS-Based Intelligent Searching Model for cross-organization identity management in Web service, The Fourth International Conference on Electronic Business (ICEB2004), Beijing, China, 05 Dec 2004.
4. An-Ping Chen, Wei-Li Lin, Yen-Chu Chen, An Intelligent Model for Stock Investment with Buffett Strategy、Classifier System、Neural Network and Linear Programming, The Fourth International Conference on Electronic Business (ICEB2004), Beijing, China, 05 Dec 2004.

在產業界經歷方面:

1992 年~1998 年任職法商 Alcatel-TAISEL 通訊系統研發工程師。

1998 年~1999 年任職 ERP 軟體商崧揚科技創始股東暨業務專案經理。

1999 年~2003 年任職德商 Siemens Telecommunication (Taiwan)業務部門經理。

2004 年~2005 年兼任軟交換系統公司 Walkersun Technologies 總經理特別助理。

2005 年~2006 年同時任職亞太電信 M 化事業群業務處、行銷處、產品開發處協理。

2006 年~2009 年任職行動美商電信網管系統公司 Groundhog Technologies 副總經理暨大中國區執行副總經理。