Fuzzy decision support system for risk analysis in e-commerce development

E.W.T. Ngai*, F.K.T. Wat

Department of Management and Marketing, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong, People's Republic of China

Received 1 December 2001; accepted 1 December 2003
Available online 14 March 2004

Abstract

This paper describes the development of a fuzzy decision support system (FDSS) for the assessment of risk in e-commerce (EC) development. A Web-based prototype FDSS is designed and developed to assist EC project managers in identifying potential EC risk factors and the corresponding project risks. A risk analysis model for EC development using a fuzzy set approach is proposed and incorporated into the FDSS. The results of an evaluation indicate that the prototype performs to expectations.

© 2004 Elsevier B.V. All rights reserved.

Keywords: Electronic commerce; Fuzzy decision support system; Fuzzy set; Risk analysis

1. Introduction

E-commerce (EC) is “a modern business methodology that addresses the needs of organizations, merchants, and consumers to cut costs while improving the quality of goods and services and increasing the speed of service delivery. The term also applies to the use of computer networks to search and retrieve information in support of human and corporate decision making” [21]. It has been adopted widely in most enterprises.

Although EC offers various business opportunities, EC development is plagued by various kinds of risk and risk management is necessary to avoid these problems. Indeed, a task that is critical to the proper management of EC development is the assessment of risk. An important step in advancing our knowledge requires that we understand and address these risks. According to Leung et al. [26], most project managers worry about the time involved in risk management when it comes to identifying and assessing risks. However, with the aid of computers and the use of software systems, the time for risk analysis can be significantly reduced. Risk analysis can be conducted by using the theory of probability, which estimates the likelihood and consequences of any given risk. EC development is relatively new to most companies, and only limited information is available on the associated risks. The application of fuzzy set theory (FST) to risk analysis seems appropriate, as such analysis is highly subjective and related to inexact and vague information. There is a need to design and develop a fuzzy decision support system (FDSS) to assist EC practitioners to evaluate the risks associated with EC development.
This paper describes the research and development of a FDSS that can be used to effectively support EC project managers in conducting risk assessment in EC development. The motivation for the present work is the recognized absence and need for such a system that help in the evaluation of a company’s risk level and provides an overall risk evaluation of EC development.

2. Literature review

2.1. Definitions of risks associated with EC development

The concept of “risk” became popular in economics during the 1920s. Since then, it has been successfully used in theories of decision making in economics, finance, and the decision science. The Merriam-Webster [48] dictionary defines risk as the “possibility of loss or injury” or “someone or something that creates or suggests a hazard”. At present, there is no agreed upon universal definition of EC risk but information security is a widely recognized aspect of EC risk [44]. Greenstein [17] views risks associated with EC as the possibility of loss of confidential data or the destruction, generation, or use of data or programs that physically, mentally or financially harms another party, as well as the possibility of causing harm to hardware. Mceachern [29] uses the term “cyber risk” to define any risk associated with EC, including, for example, Web site destruction and manipulation, unauthorized access to customer records, Internet fraud, telecommunications theft, copyright infringement and denial of access. On the other hand, Viehland [44] focuses on managing business risk in EC. He defines EC risk as the likelihood of a negative impact to organization itself when developing or operating EC strategy. In this paper, risks associated with EC development are the risks of direct or indirect loss to the organization in development an EC project, which refers to any project that involves development stages as planning, analysis, design and implementation of an EC system.

2.2. The significance of fuzzy risk analysis for EC development

Through using EC, companies are able to connect with their trading partners for “just in time production” and “just in time delivery”, which improves their competitiveness globally. Although EC offers great opportunities, there is no doubt that EC development involves many risks. In this study, we do not intend to present risks to EC only but also risks that EC development shares with traditional systems. Every EC development is linked to a different degree of risk. However, most companies do not identify and assess EC-related risk. EC development has a lot in common with IT project development. Many IT and EC development cannot be completed on-time and on-budget [39]. Proper risk management is an essential element of project success [39] because without appropriate risk management it fails to achieve significant return on investment or defensive/competitive purpose. One of the important phases in risk management is risk analysis, which involves a process of risk identification and risk assessment. Proper risk assessment can enhance the chance of successful project implementation [1]. McDonald [28] and Stoehr [39] point out that companies need to perform a risk analysis before engaging EC development.

2.3. Fuzzy risk analysis research

The techniques of risk analysis are powerful tools to help people manage uncertainty. Thorough risk analysis estimation and evaluation can provide valuable support for decision making. There are many risk analysis techniques currently in use that attempt to evaluate and estimate risk. These techniques can be either qualitative or quantitative depending on the information available and the level of detail that is required [4]. Quantitative techniques rely heavily on statistical approaches, which include Monte Carlo Simulation [49], Fault and Event Tree Analysis [4,49], Sensitivity Analysis [49], Annual Loss Expectancy [35], Risk Exposure [5], Failure Mode and Effects Analysis [49], etc. On the other hand, qualitative techniques rely more on judgment than on statistical calculations such as Scenario Analysis [35], FST [35], etc. Quantitative and qualitative techniques have their own advantages and disadvantages. Among these techniques, the application of FST to risk analysis seems appropriate, as such analysis is highly subjective and related to inexact and vague information.

Since FST was introduced by Zadeh [51] to deal with problems in which vagueness was present, lin-
gustic values have been widely used to approximate reasoning. Numerous studies of FST in risk assessment have appeared in different areas, and are summarized in Table 1. FST has been effective in such a variety of areas because it can handle inexact yet useful information.

2.4. Fuzzy weighted average

An operation commonly used in risk and decision analysis is the weighted average operation [20], which takes the following form:

\[
\bar{W} = \frac{\sum_{i=1}^{n} W_i \times R_i}{\sum_{i=1}^{n} W_i}
\]

where \( \bar{W} \) is the weighted average of ratings, \( R_i \) is the rating according to criterion \( i \), and \( W_i \) is the weight assigned to criterion \( i \). When the terms \( R_i \) and \( W_i \) are represented by fuzzy sets or fuzzy numbers, the above operation is referred to as a fuzzy weighted average (FWA). Bass and Kwakernaak [3] were amongst the earliest researchers to determine the ranking of multiple alternatives with this weighted average. Later, Schmucker [37] used the FWA to propose an approximate numerical method known as the “Fuzzy Risk Analyzer”. Furthermore, many applications such as Refs. [40,50] follow Schmucker’s procedure, and it is widely applied in risk analysis, particularly in relation to construction projects. However, Dong et al. [13] have shown that Schmucker’s discretization method can give quite irregular and incorrect membership functions because information is lost in the process of ensuring its convexity. They introduced the DSW
algorithm [13] as an alternative, and dealt with the problem through interval analysis and alpha-cut representations of fuzzy sets. Later, Dong and Wong [12] employed a simple, efficient, and systemic method to develop an FWA algorithm based on the DSW algorithm. The FWA algorithm has been widely adopted in civil engineering, especially in damage assessment such as [32,36,41].

However, as this algorithm requires $O(2^n)$ comparisons and arithmetic operations, Liou and Wang [27] suggested an improved fuzzy weighted average algorithm (IFWA), whilst Lee and Park [24] proposed an efficient fuzzy weighted average algorithm (EFWA). In the present study, a risk assessment model based on the fuzzy weighted average of FST will be used to calculate the overall risk faced by EC projects. The EFWA algorithm is used because it can reduce the number of comparisons and arithmetic operations to $O(n \log n)$ rather than $O(n^2)$, as is the case with the IFWA [27].

3. System development methodology for the FDSS

The purpose of this study is to design and develop a FDSS to assist EC project managers in identifying potential risk factors and evaluating the corresponding EC development risks. FDSS is constructed following the five-stage system development methodology, which is based on a generic IS development [31], incorporated with the method for fuzzy risk analysis [37,40,41,46]. Although this system development methodology is developed for the FDSS, we believe that other researchers can easily follow as a guideline to design and develop other FDSS for risk analysis in other application areas. The system development process consists of five stages, namely, construction of fuzzy risk analysis model, development of system architecture, analyzing and designing of the system, building of the prototype, and evaluation of the system. An overview of these five stages of system development is shown in Fig. 1. First, a fuzzy risk analysis model was constructed as the kernel of the system. Second, system architecture was developed. Third, system design and analysis were carried out in modularity with defining functionalities of the system components and understanding of how they interact with one. Fourth, the prototype system was built in

![Fig. 1. FDSS development methodology framework.](image-url)
order to learn more about the concepts, framework, and design through the system-building process. Finally, the prototype system was evaluated by EC experts and potential users. Detailed descriptions of each phase are presented in the following sections.

3.1. Phase 1: construct a fuzzy risk analysis model

Most existing risk analysis models are based on quantitative techniques such as Monte Carlo Simulation and Annual Loss Expectancy. However, the information that is related to most uncertainty factors is not numerical. FST provides an approximate model for the evaluation of the risk faced by EC projects through a linguistic approach. The procedure for fuzzy risk analysis is based on the works from Refs. [37,40, 41,46] that consisted of five steps: risk identification, natural language representation, fuzzy assessment aggregation, fuzzy weighted average computation, and linguistic approximation. The details of each stage are described in the following sections.

### 3.1.1. Risk identification

The first step is to conduct risk identification and compile a list of the most significant uncertainty factors and their descriptions. Before conducting fuzzy risk analysis, one must identify the components of risks associated with EC development. However, little empirical research has focused on identifying the potential risk factors that threaten EC development. In the study of Wat et al. [47], a source-based approach to categorizing EC development risks is initially used, with technical, organizational, and environmental risks as three primary source categories. Then the potential risks associated with EC development was identified with 51 risk items (see Table 2) associated with EC development based on a comprehensive literature review and interviewed with EC practitioners.

An empirical study was conducted with 330 valid returns used for the analysis. An exploratory factor analysis (EFA) of the survey data revealed 10 major

<table>
<thead>
<tr>
<th>Variables</th>
<th>Potential risks associated with EC development</th>
</tr>
</thead>
<tbody>
<tr>
<td>V1</td>
<td>Hacker gaining unauthorized access</td>
</tr>
<tr>
<td>V2</td>
<td>Absence of firewall</td>
</tr>
<tr>
<td>V3</td>
<td>Lack of using cryptography</td>
</tr>
<tr>
<td>V4</td>
<td>Poor “key” management</td>
</tr>
<tr>
<td>V5</td>
<td>Malicious code attacks</td>
</tr>
<tr>
<td>V6</td>
<td>Disclosure of sensitive information</td>
</tr>
<tr>
<td>V7</td>
<td>Loss of audit trail</td>
</tr>
<tr>
<td>V8</td>
<td>Natural disaster-caused equipment failure</td>
</tr>
<tr>
<td>V9</td>
<td>Human factor-caused equipment failure</td>
</tr>
<tr>
<td>V10</td>
<td>Threat of sabotage in internal network</td>
</tr>
<tr>
<td>V11</td>
<td>Inadequate backup systems</td>
</tr>
<tr>
<td>V12</td>
<td>Software or hardware problem-caused system failure</td>
</tr>
<tr>
<td>V13</td>
<td>Site or network overload and disruption</td>
</tr>
<tr>
<td>V14</td>
<td>Poor design, code or maintenance procedure</td>
</tr>
<tr>
<td>V15</td>
<td>Wrong functions and properties development</td>
</tr>
<tr>
<td>V16</td>
<td>Wrong user interface development</td>
</tr>
<tr>
<td>V17</td>
<td>Project complexity</td>
</tr>
<tr>
<td>V18</td>
<td>Wrong project size estimation</td>
</tr>
<tr>
<td>V19</td>
<td>Technological newness</td>
</tr>
<tr>
<td>V20</td>
<td>Continuous change of system requirements</td>
</tr>
<tr>
<td>V21</td>
<td>Wrong schedule estimation</td>
</tr>
<tr>
<td>V22</td>
<td>Project behind schedule</td>
</tr>
<tr>
<td>V23</td>
<td>Project over budget</td>
</tr>
<tr>
<td>V24</td>
<td>Inadequate cash flow</td>
</tr>
<tr>
<td>V25</td>
<td>Personnel shortfalls</td>
</tr>
<tr>
<td>V26</td>
<td>Lack of expertise and experience in e-commerce</td>
</tr>
<tr>
<td>V27</td>
<td>Loss of key person</td>
</tr>
<tr>
<td>V28</td>
<td>Lack of top management support</td>
</tr>
<tr>
<td>V29</td>
<td>Poor project planning</td>
</tr>
<tr>
<td>V30</td>
<td>Unclear project objectives</td>
</tr>
<tr>
<td>V31</td>
<td>Indefinite project scope</td>
</tr>
<tr>
<td>V32</td>
<td>Lack of contingency plans</td>
</tr>
<tr>
<td>V33</td>
<td>Business process redesign</td>
</tr>
<tr>
<td>V34</td>
<td>Organizational restructuring</td>
</tr>
<tr>
<td>V35</td>
<td>Lack of trust between your organization and merchant or customer</td>
</tr>
<tr>
<td>V36</td>
<td>Inappropriate media for the product and service</td>
</tr>
<tr>
<td>V37</td>
<td>Lack of international legal standards</td>
</tr>
<tr>
<td>V38</td>
<td>New laws, regulations, and judicial decisions constantly change the online legal landscape</td>
</tr>
<tr>
<td>V39</td>
<td>Uncertain legal jurisdiction</td>
</tr>
<tr>
<td>V40</td>
<td>Incompletion of contract terms</td>
</tr>
<tr>
<td>V41</td>
<td>Difficult to change outsourcing decision/vendor</td>
</tr>
<tr>
<td>V42</td>
<td>Loss of data control</td>
</tr>
<tr>
<td>V43</td>
<td>Loss of control over vendor</td>
</tr>
<tr>
<td>V44</td>
<td>Loss of control over information technology</td>
</tr>
<tr>
<td>V45</td>
<td>Hidden cost</td>
</tr>
<tr>
<td>V46</td>
<td>Lack of vendor expertise and experience</td>
</tr>
<tr>
<td>V47</td>
<td>Lock-in situation</td>
</tr>
<tr>
<td>V48</td>
<td>Vendor offers outdated technology skill</td>
</tr>
<tr>
<td>V49</td>
<td>Vendor provides poor quality service</td>
</tr>
</tbody>
</table>

Table 2 (continued)

<table>
<thead>
<tr>
<th>Variables</th>
<th>Potential risks associated with EC development</th>
</tr>
</thead>
<tbody>
<tr>
<td>V50</td>
<td>Difference users with different in culture customers, and business styles</td>
</tr>
<tr>
<td>V51</td>
<td>Language barrier</td>
</tr>
</tbody>
</table>

Source: Based on Wat et al. [43].
dimensions of risks associated with EC development, namely: (1) resources risk, (2) requirements risk, (3) vendor quality risk, (4) client–server security risk, (5) legal risk, (6) managerial risk, (7) outsourcing risk, (8) physical security risk, (9) cultural risk, and (10) re-engineering risk (see Table 3). As a result of the study [47], the risk classification framework as shown in Fig. 2 helps in the formulation of ways of accessing risks to EC development.

3.1.2. Natural language representation

According to Karwowski and Mital [22], traditional approaches to risk assessment obtain their overall risk scores by calculating the product of exposure, likelihood, and the consequences of a possible accident due to the hazard. A simpler approach that is advocated by some risk experts is to multiply the severity of consequences by the likelihood of their occurrence, as the likelihood of occurrence automatically includes exposure [45]. For example, Boehm [5] defined risk impact as the product of the probability of an unsatisfactory outcome (Likelihood) and the loss to the parties affected when the outcome is unsatisfactory (Severity). Consequently, two linguistic variables, “Likelihood” and “Severity”, are defined to calculate the overall risk. In FWA, “Likelihood” is the rating factor \((R_i)\), and “Severity” is the weighting factor \((W_i)\) that corresponds to rating factor \(i\). Both linguistic variables have five terms, “Very Unlikely”, “Unlikely”, “Medium”, “Likely”, and “Very Likely” (see Fig. 3). “Severity” is expressed as “Minimal”, “Low”, “Moderate”, “High”, and “Critical” (see Fig. 4).

In this study, the membership functions of the linguistic terms are characterized by triangular fuzzy numbers, as these are very often used in applications such as fuzzy controllers, and in managerial decision making, business and finance, and the social sciences, etc. [6]. Table 4 shows the membership functions and the triangular fuzzy numbers of each linguistic term.

3.1.3. Fuzzy assessment aggregation

In this stage, an aggregate of several evaluators’ fuzzy assessment is performed by using the fuzzy average operation for aggregate method. By allowing more than one evaluator to assess the risks associated with an EC project, a more objective and unbiased result can be obtained. The fuzzy average operation for aggregate method that is known as the “Triangular Average Formula” [6] is used to determine the mean of evaluator opinions. Hence, the fuzzy average of each risk factor question from the risk assessment form can be obtained. The Triangular Average Formula is as follows:

| Factor 1: | V21 | 0.640 | 11.567 | 26.900 | 26.900 |
| Factor 2: | V14 | 0.535 | 3.050 | 7.094 | 33.994 |
| Factor 3: | V46 | 0.679 | 2.349 | 5.463 | 39.457 |
| Factor 4: | V1 | 0.630 | 2.034 | 4.730 | 44.187 |
| Factor 5: | V37 | 0.786 | 1.608 | 3.739 | 47.926 |
| Factor 6: | V28 | 0.522 | 1.451 | 3.375 | 51.302 |
| Factor 7: | V40 | 0.539 | 1.343 | 3.124 | 54.426 |
| Factor 8: | V7 | 0.519 | 1.197 | 2.785 | 57.210 |
| Factor 9: | V50 | 0.804 | 1.112 | 2.586 | 59.796 |
| Factor 10: | V33 | 0.606 | 1.075 | 2.501 | 62.297 |

Table 3 Results of EFA
Fig. 2. A classification framework for risk management in EC development.
Consider \( n \) evaluators and let \( A_i = (a^{(1)}(i), a^{(M)}(i), a^{(2)}(i)) \) be the fuzzy numbers, \( i = 1, \ldots, n \). Thus, the triangular average mean:

\[
A_{\text{average}} = \frac{1}{n} \sum_{i=1}^{n} A_i = \frac{1}{n} \left( \sum_{i=1}^{n} a^{(1)}(i), \sum_{i=1}^{n} a^{(M)}(i), \sum_{i=1}^{n} a^{(2)}(i) \right)
\]

### 3.1.4. Fuzzy weighted average computation

Based on the classification framework in Fig. 2, overall EC project risk has three dimensions, which consist of 10 components. The lowest-level node encompasses a list of risks factors that are associated with EC development. Having obtained the fuzzy average with the likelihood and severity of each risk derived from previous step, the EFWA algorithm (see Figs. 3 and 4) can be used:

\[
A_{\text{average}} = (m_1, m_m, m_2)
\]

Fig. 3. Membership function of likelihood.

Fig. 4. Membership function of severity.
Appendix A) can be applied. For a better understanding, let us consider an example. Fig. 5 illustrates a simple hierarchical structure that is designed to identify the severity of loss for “Legal Risk”. There are three risk factors associated with “Legal Risk”, and the fuzzy weighted average for the severity of loss is calculated as follows:

$$\bar{W} = \frac{W_1 \times R_1 + W_2 \times R_2 + W_3 \times R_3}{W_1 + W_2 + W_3}$$

A numerical illustration of this calculation is shown in Appendix B.

### 3.1.5. Linguistic approximation

As the result of the calculated fuzzy weighted average is a fuzzy number, it is necessary to translate it back into linguistic terms for easy interpretation. The goal of linguistic approximation is to find the linguistic term with the closest possible meaning to that of a defined fuzzy set. There are three techniques in linguistic approximation: best fit, successive approximation, and piecewise decomposition. The difference between these three techniques was discussed by Schmucker [37]. In the present study, the best fit method is adopted because it is easy to understand and easy to implement on computers [37]. This method is based on the “Euclidean distance” between two fuzzy sets, as proposed by Dubois and Prade [14]. Euclidean distance is defined as:

$$d(X, A) = \left\{ \sum_{i=1}^{n} [X(i) - A(i)]^2 \right\}^{1/2}$$

$d=$Euclidean distance between two fuzzy sets, $X=$resultant fuzzy set, and $A=$defined fuzzy set.

However, it is necessary to map the resultant fuzzy interval back to one of the fuzzy numbers that are defined in Fig. 4. Therefore, a modified Euclidean approach was proposed by Ross et al. [36]. The difference measure $d$ is given as follows:

$$d(X, A) = \left( \sum_{i=1}^{n} \left( X_{\text{min}}(i) - A_{\text{min}}(i) \right)^2 + \left( X_{\text{max}}(i) - A_{\text{max}}(i) \right)^2 \right)^{1/2}$$

where $i$ is the $z$ value and $n$ is the number of $z$-cut.

### Table 4

<table>
<thead>
<tr>
<th>Likelihood</th>
<th>Severity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Very unlikely minimal</td>
<td>$(0, 0, 0.25)$</td>
</tr>
<tr>
<td>Unlikely low</td>
<td>$(0.25, 0.5)$</td>
</tr>
<tr>
<td>Medium moderate</td>
<td>$(0.25, 0.75)$</td>
</tr>
<tr>
<td>Likely high</td>
<td>$(0.5, 1)$</td>
</tr>
<tr>
<td>Very likely critical</td>
<td>$(0.75, 1)$</td>
</tr>
</tbody>
</table>

### Fig. 5

Simple hierarchical structure of legal risk.
Applying the above equation, the Euclidean distance between fuzzy set \( X \) and predefined natural language expressions (e.g. low, moderate, and high) is calculated. The model then assigns the appropriate natural language expression to the lowest Euclidian distance associated with fuzzy set \( X \). For example, if fuzzy set \( X \), which represents the total risk faced by an EC project, has a low Euclidian distance with a predefined natural language expression of “high”, then the model assumes that fuzzy set \( X \) is translated as “high”, which indicates that the total risk faced by the project is high. Refer to Appendix B for detailed numerical illustration of the application to the example of “Legal Risk”.

3.2. Phase 2: develop system architecture

Good system architecture provides a road map for the system building process by placing components into perspective, defining their functionalities, and demonstrating how they will interact with one another [31]. The Web is the center of activity in developing decision support systems (DSS) [38] while client–server architecture has been widely adopted in the integration of Web-based applications [9]. The client–server relationship describes the distribution of tasks between a server and the clients who access that server. The FDSS is a client–server system with a two-tiered architecture. On the client side it is a front-end system that works with Web clients to obtain service requests and present results. On the server side, it is a back-end system that executes a fuzzy risk analysis and access database for data management. Indeed, such a two-tiered architecture is suitable when developing non-critical applications with light transaction loads such as DSS or departmental applications [11]. Since the FDSS is a client–server system, it will be executed on the Web server. Whenever a Web browser (Client) sends a request for a page to the FDSS, the code is processed at that time by the Web server. For the system components contain in the FDSS, it is composed of three interrelated components, which are (1) database, (2) model base subsystem, and (3) user interface. These three components are the basic elements in DSS [33]. Fig. 6 depicts the basic architecture of the FDSS.

3.3. Phase 3: analyze and design the system

Analysis and design are important aspects of the system development process. Design involves an understanding of the domain being studied, the application of various alternatives, and the synthesis and evaluation of proposed solutions. Design specifications are used as a blueprint for the implementation of the system [31]. The determination of system components and development platform is made during this phase. The design of DSS can be divided into three interrelated components, which are database, model base subsystem, and user interface [33]. The detailed specifications of these three system components, structure, and features are determined as follows.

3.3.1. Database

The database system is responsible for the storage of data and its management. It maintains the necessary information on each EC project. The data is obtained from an external source through manual or automated processes and the results generated by the FDSS. To manipulate databases on the Web, ActiveX Data Object (ADO) is used to interface with relational databases via the Open Database Connectivity (ODBC) protocol [2]. ADO was chosen as the data access mechanism due to its high speed, ease of use, and low memory overheads. The underlying database can be any application that supports the ODBC protocol. The current implementation makes use of Microsoft Access.

3.3.2. Model base subsystem (fuzzy risk analysis COM component)

The model base performs activities to provide analytical capabilities for the DSS [43]. Users can write their own models or use standard models at times. Fuzzy risk analysis model described in Section 3.1 is employed as a model base subsystem in FDSS. This model is translated into programming code and is integrated as the Component Object Model (COM). COM defines the binary interface between objects. It is a binary interoperability specification. The two most common reasons for using components are breaking up complex applications into manageable chunks and packaging code for re-use [2]. ASP scripting is mainly used to implement the FDSS. ASP script has the ability to interface with COM
Fig. 6. System architecture.
compliant software components. If functionality is needed but cannot be provided by scripting, then ASP components can be used. ASP components are COM-based, encapsulate a specific functionality, and are invoked either directly from an ASP page or indirectly via another ASP component [34].

Fuzzy risk analysis is implemented as the COM object that is stored in DLL for performing fuzzy risk analysis. When clients invoke the calculation of the overall risk faced by the EC project, the fuzzy risk analysis COM component is called to access necessary information from the database, such as the likelihood and severity of each risk factor, to perform fuzzy averaging, calculate the fuzzy weighted average, and obtain linguistic approximations. Eventually, the overall risk and risk score of each risk dimension are obtained.

3.3.3. User interface

The design of the user interface is a key element in DSS functionality. The DSS interface should provide easy communication between the user and the system [43]. Web browser serves as the user interface component of the DSS, which make the technology easy to understand and use [38]. Besides, the FDSS consists mainly of menus and graphics, which are supplemented by natural language. A client invokes the system by connecting to the Web site through the standard HTTP protocol, which causes the interface component to be loaded from the server to the client station. Pull-down menus allow users to specify their needs, such the creation of a new project and the addition of an evaluator record (see Fig. 7).

3.4. Phase 4: build the prototype system

The implementation of a system demonstrates the feasibility of the design and the utility of the functionalities that are envisaged [31]. Building a prototype system is one of the processes that allow insight into the problems and the complexity of a system during development research. FDSS is constructed using various commercial software packages and programming techniques. The detail descriptions of construction of the FDSS using these software packages and programming skills are given as below.

The prototype was run on the Windows 2000 Server™ platform. IIS was the Web server which accepted the ASP request forms that were sent from client browsers. Internet Explorer 5.0 was selected as the Web browser in the client computers. Macromedia Dreamweaver™ was selected as the HTML editor for Web site and page design, and it was incorporated with Macromedia Fireworks™ for the creation and editing of images, and Macromedia Flash™ for the creation of animated openings.

The main body of the FDSS was written using ASP script language. VBScript was chosen for the server-side script because it is the default scripting language on the IIS server. To create more functional and interactive Web pages, JavaScript was selected as the client-side script because it is widely supported, and some browsers (e.g. Netscape Communicator 4.5) do not have the capability to interpret VBScript except with the aid of proprietary add-ons from third-party vendors. In the FDSS, JavaScript embedded in ASP code was mainly responsible for HTML form and data validation, and pull-down menus were used in the interface design. Microsoft Visual InterDev (VI) was also used to design dynamic Web applications, as it provides a development environment and collection of useful tools and utilities [9].

Visual Basic™ was selected for development of the fuzzy risk analysis component. According to Anderson et al. [2], there are several advantages of using Visual Basic™ to develop COM components. Firstly, it allows the user to write components quickly without having to spend much time learning COM. Secondly, it gives quite respectable performance. Thirdly, it is often seen as the natural progression for ASP developers learning to write components. Power [34] argues that the most common, and the simplest, component to create is Visual Basic, especially using an ActiveX DLL that is an in-process component.

The ODBC protocol was selected to communicate with the Access database because it is compatible with a variety of database systems. ADO was used to provide the database connection, as it is a set of objects that allows programmers to program their data access logic in languages such as Visual Basic, as well as in scripting languages [9].

3.5. Phase 5: evaluate the system

Once the system is developed, the testing and evaluation of the prototype can be performed.
Through system evaluation, information can be captured on what users like and dislike, and what the system does and does not do to meet their needs. Firstly, testing and evaluation of the system were performed. All of the FDSS modules were tested for accuracy and completeness, and the outputs generated were checked and validated. These tests ensured that the system was performing functions that would meet the requirements of users by assisting them in conducting risk management for EC development. Secondly, once the FDSS was built, outcome evaluation was conducted in two phases. The first phase was domain expert evaluation, and the second phase was potential user (EC practitioner) evaluation.

There are a number of approaches to evaluate DSS. One of the criteria for the evaluation of a DSS is the measurement of the effectiveness of the system. Another evaluation criterion is to measuring user satisfaction. An evaluation form with several sections was designed. The first section measured the effectiveness and usability of the system with five-point Likert scales (1=strongly disagree, 3=undecided, 5=strongly agree). Through measuring the effectiveness of the system, we can see the ability of the system to accomplish its objectives or mission. Items to measure the usability of the system reflect the usefulness and ease of use of the system. We can therefore assess user satisfaction as one of the potential indicators of the system’s success. The second
section of the evaluation form included several open-ended questions that were analogous to an interview in that they gave the respondents an opportunity to express themselves openly, particularly about the problems that they encountered and how the prototype could be improved. The final section collected the evaluators’ personal information.

3.5.1. Expert evaluation

Evaluations by domain experts help to determine the accuracy of embedded knowledge [16]. The FDSS system was validated by a group of six participants attending the “Fuzziness and Soft Computing in the New Millennium” session of the 9th International Fuzzy System Association World Congress and the 20th North American Fuzzy Information Processing Society International Conference. The FDSS was demonstrated and an evaluation form was distributed to the experts who presented papers in the fuzzy modeling session of the meeting. All experts were university professors and researchers with average more than 10 years working experience and good knowledge of FST. They were asked to evaluate the prototype from two perspectives: effectiveness and usability of the FDSS. For the effectiveness of the system, most respondents agreed that the prototype system was an effective risk assessment tool, in particular indicating that it can assist in assessing risks associated with EC development. For the usability of the system, most respondents considered the system quite easy to use. User’s interaction with the system was clear and understandable. They were likely to recommend the prototype system to other users.

3.5.2. Potential user evaluation

Evaluations by users help to determine the utility of a system according to the following criteria: ease of interaction, the extent of its capabilities, its efficiency and speed, its reliability and whether it produces useful results [16]. The evaluation form was transformed electronically and was integrated in the FDSS for user to conduct online evaluation. A total of 50 e-mails were sent to a randomly selected sample from our mailing list directory of EC practitioners in Hong Kong. Twenty-two responses were collected and stored in the FDSS database for further analysis.

The responses from the potential users are summarized in Table 5. The potential users rated the system highly on “effectiveness of the system” and “usability”, with a mean score of at least 3.14 on a five-point scale (1=strongly disagree, 3=undecided, 5=strongly agree). Based on the results of the evaluation, the prototype was seen as a promising system for the management of risk in EC development.

In order to ensure the values of most the mean responses were statistically significantly different from the neutral value of the scale, which is “3—undecided”, one-sample t-test using test value “3” was conducted for the 10 items. Results of t-test are presented in Table 5. Overall, the prototype evalu-

<table>
<thead>
<tr>
<th>Table 5</th>
<th>Result of prototype evaluations by potential users</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mean</td>
</tr>
<tr>
<td>Effectiveness of the system</td>
<td></td>
</tr>
<tr>
<td>The system can</td>
<td></td>
</tr>
<tr>
<td>(1) Assist in assessing risks associated with EC development.</td>
<td>3.59</td>
</tr>
<tr>
<td>(2) Provide an effective mean to collect, store and analyze perception on potential risk to EC development.</td>
<td>3.64</td>
</tr>
<tr>
<td>(3) Monitor and mitigate risk.</td>
<td>3.55</td>
</tr>
<tr>
<td>Usability of the system</td>
<td></td>
</tr>
<tr>
<td>(4) Learning to operate the system would be easy for me.</td>
<td>3.32</td>
</tr>
<tr>
<td>(5) My interaction with the system would be clear and understandable.</td>
<td>3.86</td>
</tr>
<tr>
<td>(6) I find the system to be flexible to interact with.</td>
<td>3.18</td>
</tr>
<tr>
<td>(7) The system’s commands are self-explained and easy to understand.</td>
<td>4.05</td>
</tr>
<tr>
<td>(8) I find the system easy to use.</td>
<td>3.14</td>
</tr>
<tr>
<td>(9) The system is user friendly.</td>
<td>4.09</td>
</tr>
<tr>
<td>(10) Likely to recommend to other users.</td>
<td>3.59</td>
</tr>
</tbody>
</table>

* p≤0.05.
** p≤0.01.
*** p≤0.001.
4. Benefits of using FDSS

FDSS had been implemented and the results of the system evaluation showed that FDSS can be applied effectively for managing risks associated with EC development. The computations involved in the model of fuzzy risk analysis are tedious if performed manually. It is an easy task and the time for risk analysis can be significantly reduced. The Web-based FDSS automates a questionnaire instrument for risk assessment that helps the EC project managers to determine the overall risk of EC development. The benefits of using the system are as follows.

- Risks associated with EC development are identified. These risk items serve as a checklist that cover possible risks associated with EC development in technical, organizational, and environmental dimensions. EC project managers or EC practitioners can be informed and be able to recognize the risks associated with EC development.
- EC project managers can predict the overall risk of the project before start the implementation. An overall risk index can be used as early indicators of project problems or potential difficulties. Evaluators can keep track to evaluate the current risk level of their EC development.
- The system provides an effective, systematic, and more natural way by using the proposed fuzzy risk analysis model. Evaluators can just simply use the risk evaluation checklist and use the linguistic terms to evaluate the EC development risk level.
- Prioritization is necessary to provide focus for important risks [18]. A list of ranked risk items associated with EC development will be produced. Therefore, the most serious risk item will be addressed first.

5. Conclusions and further enhancements

EC development takes place in a complex and dynamic environment that includes high levels of risk and uncertainty. This study has outlined an approach to the assessment of the risks associated with EC development using FST. A model of fuzzy risk analysis was proposed to assist EC project managers and decision makers in formalizing the types of thinking that are required in assessing the current risk environment of their EC development in a more systematic manner than before.

A Web-based FDSS was designed and developed to incorporate the proposed risk analysis model. System evaluation was performed to ascertain whether the FDSS achieved its designed purpose, and the results were satisfactory. The feedback and comments collected from respondents were used to make necessary adjustments. The results of the evaluation strongly support the viability of the study approach to risk analysis using fuzzy sets, and demonstrated the feasibility of evaluating EC project risk.

The FDSS prototype focused on risk identification, analysis, and prioritization. Less attention was given to the risk management planning, resolution, and monitoring that is associated with EC development. Further research should be conducted into such risk management planning. In addition, risk monitoring should be conducted regularly to track the status of the identified risks. With such insight and improvement, the FDSS could be further enhanced to handle the functionality of risk management.

Moreover, it was assumed that the “weighting” assigned by each evaluator in the risk evaluation was the same, but the relative importance placed on certain factors by individual decision makers and experts could be widely different. Further research is needed to develop different “weightings” for different evaluators.

6. Limitations of this study

Although the prototype comes out with many advantages, it still has some limitations. The limitations of the prototype are summarized below.
• In spite of the fact that the prototype evaluation shows a satisfactory outcome in the effectiveness and usability of the prototype, but FDSS do not get the chance to test it with real-life EC projects. The validity of the system can be established through in-depth case studies.
• The prototype only provides the risk items based on the risk classification framework shown in Fig. 2. The list of risks shown in Table 2 is not exhaustive, but it is comprehensive enough for the purpose of this study.
• For simplification, the membership functions were evenly distributed by triangular fuzzy numbers. Various membership functions need to be estimated to be as realistic as possible.

Acknowledgements

The authors are grateful for the constructive comments of the referees and the Area Editor on an earlier version of this paper. They wish to thank all the people who have participated in the evaluation of the system as well.

Appendix A. Algorithm procedure of EFWA (source: Ref. [24])

1. Sort a’s in non-decreasing order. Let \( (a_1, a_2, \ldots, a_n) \) be the resulting sequence. Let \( \text{first}=1 \) and \( \text{last}=n \).
2. Let \( \delta_{\text{threshold}}=|(\text{first}+\text{last})/2| \). For each \( i=1, 2, \ldots, \delta_{\text{threshold}} \), let \( e_i=d_i \) and for each \( i=\delta_{\text{threshold}}+1, \ldots, n \), let \( e_i=c_i \). For an \( n \)-tuple \( S=(e_1, e_2, \ldots, e_n) \), evaluate \( \delta_{S_{\text{threshold}}} \) and \( \delta_{S_{(\delta_{\text{threshold}}+1)^{\text{threshold}}}} \).
3. If \( \delta_{S_{\text{threshold}}}>0 \) and \( \delta_{S_{(\delta_{\text{threshold}}+1)^{\text{threshold}}}}\leq 0 \) then \( L=f_L(e_1, e_2, \ldots, e_n) \) and goto Step 4; otherwise execute the following step.
   3.1. If \( \delta_{S_{(\delta_{\text{threshold}}+1)^{\text{threshold}}}} > 0 \) then \( \text{first}=\delta_{\text{threshold}}+1 \); otherwise \( \text{last}=\delta_{\text{threshold}} \), and goto Step 2.
4. Sort b’s in non-decreasing order. Let \( (b_1, b_2, \ldots, b_n) \) be the resulting sequence. Let \( \text{first}=1 \) and \( \text{last}=n \).
5. Let \( \zeta_{\text{threshold}}=|(\text{first}+\text{last})/2| \). For each \( i=1, 2, \ldots, \zeta_{S_{\text{threshold}}+1} \), let \( e_i=c_i \) and for each \( i=\zeta_{S_{\text{threshold}}}+1, \ldots, n \), let \( e_i=d_i \). For an \( n \)-tuple \( S=(e_1, e_2, \ldots, e_n) \), evaluate \( \zeta_{S_{\text{threshold}}} \) and \( \zeta_{S_{(\zeta_{S_{\text{threshold}}}+1)^{\text{threshold}}}} \).
6. If \( \zeta_{S_{(\zeta_{S_{\text{threshold}}}+1)^{\text{threshold}}}} > 0 \) and \( \zeta_{S_{(\zeta_{S_{\text{threshold}}}+1)^{\text{threshold}}}} \leq 0 \) then \( U=f_U(e_1, e_2, \ldots, e_n) \) and stop; otherwise execute the following step.
   6.1. If \( \zeta_{S_{(\zeta_{S_{\text{threshold}}}+1)^{\text{threshold}}}} > 0 \), then \( \text{first}=\zeta_{\text{threshold}}+1 \); otherwise \( \text{last}=\zeta_{\text{threshold}} \), and goto Step 5.

where

\[
\delta_{S_i} = \frac{(a_1 - a_i)e_1 + (a_2 - a_i)e_2 + \cdots + (a_n - a_i)e_n}{e_1 + e_2 + \cdots + e_n} \\
\zeta_{S_i} = \frac{(b_1 - b_i)e_1 + (b_2 - b_i)e_2 + \cdots + (b_n - b_i)e_n}{e_1 + e_2 + \cdots + e_n}
\]

Appendix B. Illustrative example of fuzzy risk analysis

Based on Fig. 5, an illustrative example is described below. Suppose that two evaluators assess an EC project. Table 6 shows the two different evaluators’ perceptions.

<table>
<thead>
<tr>
<th>Risk factors</th>
<th>Evaluator A</th>
<th>Evaluator B</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1) Lack of international legal standard</td>
<td>likely (0.5, 0.75, 1)</td>
<td>medium (0.25, 0.25, 0.5)</td>
</tr>
<tr>
<td>(2) New laws, regulations, and judicial decisions constantly change the online legal landscape</td>
<td>medium (0.25, 0.5, 0.75)</td>
<td>very likely (0.75, 1, 1)</td>
</tr>
<tr>
<td>(3) Uncertain legal jurisdiction</td>
<td>moderate (0.25, 0.5, 0.75)</td>
<td>moderate (0.25, 0.25, 0.5)</td>
</tr>
<tr>
<td></td>
<td>low (0.25, 0.25, 0.5)</td>
<td>critical (0.25, 0.5, 0.75)</td>
</tr>
</tbody>
</table>
The fuzzy average of the likelihood of risk factor 1 (lack of international legal standard) is:

\[
\frac{(0.25 + 0.5), (0.5 + 0.75), (0.75 + 1)}{2}
\]

\[
= (0.375, 0.625, 0.875)
\]

Whilst the fuzzy average of the severity of risk factor 1 is:

\[
\frac{(0.5 + 0.75), (0.75 + 1), (1 + 1)}{2}
\]

\[
= (0.625, 0.875, 1)
\]

Table 7 shows the fuzzy average of all risk factors. After obtaining the fuzzy average, the FWA can be calculated with the EFWA algorithm. Consider the three-term weighted average from Table 7,

\[
y = f(R_1, R_2, R_3, W_1, W_2, W_3)
\]

\[
= \frac{R_1 \times W_1 + R_2 \times W_2 + R_3 \times W_3}{W_1 + W_2 + W_3}
\]

Two values for \(\alpha\), viz., 0 and 1, are chosen. For \(\alpha=0\), the intervals of \(R_i\) and \(W_i\) are:

\[
[a_1 = 0.375, b_1 = 0.875], [a_2 = 0.5, b_2 = 0.875],
\]

\[
[a_3 = 0.25, b_3 = 0.75],
\]

\[
[c_1 = 0.625, d_1 = 1], [c_2 = 0.375, d_2 = 0.875],
\]

\[
[c_3 = 0.125, d_3 = 0.625]
\]

respectively for \(i=1, 2, 3\).

The computational procedure is as follows:

Step 1

Sort \(a\)'s into non-decreasing order, and the resulting sequence is

\[
[a_1 = 0.25, b_1 = 0.75], [a_2 = 0.375, b_2 = 0.875],
\]

\[
[a_3 = 0.5, b_3 = 0.875], [c_1 = 0.125, d_1 = 0.625],
\]

\[
[c_2 = 0.625, d_2 = 1], [c_3 = 0.375, d_3 = 0.875]
\]

\((a_1, a_2, a_3) = (0.25, 0.375, 0.5), first:= 1, last:= 3\)  

Step 2

\(\delta\)-threshold:=|(1+3)/2|=2, \(S=(0.625, 1, 0.375)\),

\[
\delta_{S_1} = \frac{(0.25 - 0.375) \times 0.625 + (0.5 - 0.375) \times 0.375}{0.625 + 1 + 0.375}
\]

\[
= -0.0156
\]

\[
\delta_{S_1} = \frac{(0.25 - 0.5) \times 0.625 + (0.375 - 0.5) \times 1}{0.625 + 1 + 0.375}
\]

\[
= -0.1406
\]

Step 3

As \(\delta_{S_1}<0\) and \(\delta_{S_1}<0\), execute the following step.

Step 3.1

Let \(last:=2\) and go to Step 2.

Step 2

\(\delta\)-threshold:=|(1+2)/2|=1, \(S=(0.625, 0.625, 0.375)\)

\[
\delta_{S_1} = \frac{(0.375 - 0.25) \times 0.625 + (0.5 - 0.25) \times 0.375}{0.625 + 0.625 + 0.375}
\]

\[
= 0.1058
\]

\[
\delta_{S_1} = \frac{(0.25 - 0.375) \times 0.625 + (0.5 - 0.375) \times 0.375}{0.625 + 0.625 + 0.375}
\]

\[
= -0.0192
\]
Since $\delta_S > 0$ and $\delta_S < 0$,

$L = f_l(c_1, d_2, d_3) = a_1 + \delta_S = 0.25 + 0.1058$

$= 0.3558$

Min $f_L$ is 0.3194 and go to Step 4.

Step 4

Sort $b$’s into non-decreasing order, and the resulting sequence is

$[a_1 = 0.25, b_1 = 0.75], [a_2 = 0.375, b_2 = 0.875],$
$[a_3 = 0.5, b_3 = 0.875], [c_1 = 0.125, d_1 = 0.625],$
$[c_2 = 0.625, d_2 = 1], [c_3 = 0.375, d_3 = 0.875]$

$(b_1, b_2, b_3) = (0.75, 0.875, 0.875), first:= 1, last:= 3$

Step 5

$\delta$—threshold:= $|(1+3)/2|= 2, S=(0.125, 0.625, 0.875),$

$\delta_S = \frac{(0.75 - 0.875) \times 0.125 + (0.875 - 0.875) \times 0.875}{0.125 + 0.625 + 0.875}$

$= -0.0096$

$\delta_S = \frac{(0.75 - 0.875) \times 0.125 + (0.875 - 0.875) \times 0.625}{0.125 + 0.625 + 0.875}$

$= -0.0096$

Step 6

As $\delta_S < 0$ and $\delta_S < 0$, execute the following step.

Step 6.1

Let last:=2 and go to Step 5.
Step 5
\[ \delta = \text{threshold} = \frac{(1+2)/2}{1}, \quad S = (0.125, 1, 0.875), \]
\[ \delta_{S_1} = \frac{(0.875 - 0.75) \times 1 + (0.875 - 0.75) \times 0.875}{0.125 + 1 + 0.875} = 0.1172 \]
\[ \delta_{S_2} = \frac{(0.75 - 0.875) \times 0.125 + (0.875 - 0.75) \times 0.875}{0.125 + 1 + 0.875} = -0.0078 \]
As \( \delta_{S_1} > 0 \) and \( \delta_{S_2} < 0 \),
\[ U = f_U(d_1, c_2, c_3) = b_1 + \delta_{S_1} = 0.75 + 0.1172 = 0.8672 \]
Max \( f_U \) is 0.8672 and stop.
Accordingly, the interval for \( x = 0 \) is \((0.3558, 0.8672)\), in which each point corresponds to the end
points of the triangle that represent the membership functions. The process is repeated for \( x = 1 \), and the result obtained is \((0.6417, 0.6417)\), which corresponds to the center of the triangle.
As the results are fuzzy numbers, Euclidean distances are used to map the resultant fuzzy interval back to linguistic terms.
\[ d(X, A) = \left( \sum_{i=1}^{n} [X_{\text{min}}(i) - A_{\text{min}}(i)]^2 + [X_{\text{max}}(i) - A_{\text{max}}(i)]^2 \right)^{1/2} \]
where \( i \) is the \( x \) value and \( n \) is the number of \( x \)-cut.
Based on the results, the Euclidean distances are:
\[ d(X, \text{Minimal}) = 0.9588; \quad d(X, \text{Low}) = 0.6441; \]
\[ d(X, \text{Moderate}) = 0.2122; \quad d(X, \text{High}) = 0.2240; \]
\[ d(X, \text{Critical}) = 0.5490 \]

Fig. 9. Risk assessment result.
The closest Euclidean distance is 0.2122, which means the legal risk is considered as moderate.

Appendix C. Illustrative example of the use of the FDSS for risk assessment in EC development

In this appendix, examples of a dialogue between a user and the prototype FDSS are shown. Annotations are added to give a deeper insight into the operation of the FDSS.

To conduct a risk analysis for a new project, the evaluator first creates a project evaluation record by clicking on “New” in the “Project” menu (see Fig. 7). The evaluator can then click on the Assessment button to start the fuzzy risk analysis. The risk assessment form is displayed and lists all of the identified risk factors from the database. Based on previous project development experience and their own perception, the evaluator can proceed to define the likelihood and severity of each risk factor in natural language. The first page of the risk evaluation form consists of technical risk factors. The evaluator can click on the radio button for the risk factor, “Hacker gaining unauthorized access”, to indicate its likelihood (e.g. “very unlikely”) and severity (e.g. “critical”). The evaluator can click on the risk factor for further explanation (see Fig. 8). He/she can then go to the next risk assessment form by clicking on the Next button. When all of the risk factors are evaluated, the fuzzy risk analysis COM component is invoked and the overall risk faced by the project is assessed. Fig. 9 shows the results based on inputted information and other evaluator answers for a particular project. The FDSS indicates that the overall risk for the selected project is moderate, and the score for each risk dimension is represented by an image bar.

References


[34] S. Power, Developing ASP Components, O’Reilly, California, 1999.


Mr. Francis K.T. Wat is currently a Research Associate in the Department of Management and Marketing at The Hong Kong Polytechnic University. He was awarded the degrees of BSc (Hons) and MPhil from The Hong Kong Polytechnic University. His current research interests are in the areas of e-commerce, management information systems and fuzzy systems. He has published in journal such as Information & Management, Omega, International Journal of Services Technology and Management.