An Asynchronous Group Decision Support System Study for Intelligent Multicriteria Decision Making

Patrick P. Cao and Frada V. Burstein
School of Information Management and Systems
Monash University
pcao@ponderosa.is.monash.edu.au;
frada.burstein@sims.monash.edu.au

Abstract
In modern business world group decision-making is becoming an extremely important activity. Recent research on computerized group decision support has focused extensively on supporting people co-located in place and time. There is a need, however, for more research focused on asynchronous group decision support systems (GDSS). Multicriteria decision-making (MCDM) theory has produced a wide range of techniques suitable for use in a variety of decision situations. With the support of an asynchronous GDSS which intelligently guides decision makers using an appropriate MCDM technique for a given task, the opportunity exists to explore the improvement of decision-making processes. In this paper we propose an intelligent MCDM process model assisted by an asynchronous GDSS. The paper describes a research project, which focuses on the development of an asynchronous GDSS to support MCDM processes in Lotus Notes® environment. The system also contains an intelligent component, which can guide users to select a suitable MCDM model for different problems and tasks. A system development research method is adopted in our study.

1. Introduction

Most of the work in the computerized decision support has been concerned with assisting individuals, whereas the majority of decision-making activities in modern business world take place within a group framework. Face-to-face meetings are the most natural form of group communication. The typical group decision support systems (GDSS) for face-to-face meetings are decision rooms where the participants use workstations to interact at the same time and the same place [24]. The use of such facilities, however, is usually costly in terms of both time and travel. Given the constraints of time and place, people can use other technologies such as computer conferencing or email to coordinate their work. Alternatively, an asynchronous GDSS could be designed to enable a group of experts to contribute their ideas to the decision process from different locations at any time.

GDSS will be used increasingly as a means for dispersed groups of people to work together through computer networks. At the same time, multicriteria decision-making (MCDM) theory has produced a wide range of techniques suitable to a variety of decision situation [27].

Most multicriteria decision-making processes need to be accompanied by group communications. It has been acknowledged that MCDM techniques are especially suitable for group setting [21]. To date, however, MCDM studies have concentrated on methods for individual decision makers rather than groups [11].

In addition, different MCDM techniques suit different kinds of decision situations. A quantitative approach called multifactor evaluation process (MFEP), for example, is recommended in cases where all of the important criteria can be given appropriate numerical weights and each alternative can be evaluated quantitatively in terms of these criteria [19]. However, in cases where people are not able to quantify their preferences for various criteria and alternatives, they can use an Analytical Hierarchy Process (AHP) to analyze the relative importance of criteria and further evaluate alternatives [20].
Many inexperienced users have difficulty in deciding which kind of MCDM technique will be the most suitable for their decision situation. Even for the same model, different procedures of utilization will generate different results in terms of computational efficiency and consistency [23]. Decision-makers need intelligent guidance for selecting a suitable MCDM technique within a decision support system.

Lotus Notes® is at present one of the most widely used groupware systems. It provides a range of facilities for storing data in a highly structured database format and a convenient platform for applications operated distributively. Notes, however, is limited in the types of group processes it can support [5]. There is a need for special purpose procedures to be integrated in the Lotus Notes® environment. For example, such procedures could support group decision-making processes and provide guidance in choosing MCDM techniques for a given task.

Our study deals with asynchronous GDSS, MCDM techniques, groupware (Lotus Notes®), and intelligent systems for decision support. With the support of an asynchronous GDSS, which intelligently guides decision-makers in using an appropriate MCDM model for a given task, the challenge is to discover a way to improve decision-making processes.

This paper describes a research project focusing on the implementation of an asynchronous GDSS with an intelligent component for guiding the selection of MCDM techniques. The knowledge-based component provides advice on where the range of MCDM techniques can be applied efficiently according to the characteristics of the specific decision situation. We summarize the theory and review the research in related areas; we illustrate a MCDM process model and identify how an asynchronous GDSS with an intelligent component may be used to support the whole MCDM process. Finally we explain the research method used in our study and describe the architecture of the proposed system. It will include an empirical study investigating the effects of such a system on decision-making processes.

2. Theoretical foundations

According to Dennis and Gallupe [4], empirical research in GDSS started from computer-messaging experiments and original decision support systems. The results from early experiments, however, were inconsistent – both positive and negative for the use of GDSS as tools for decision support. Later field studies, though, have shown more positive results. In these studies the GDSS used were more sophisticated, the subjects involved were more experienced and professional, the tasks performed were more complex, and observations were mostly longitudinal.

Fjermestad and Hiltz [8] provide a comprehensive study of 140 controlled experiments on GDSS research. They developed an integrated theoretical framework consisting of four major categories of variables: contextual or independent variables; intervening variables; group adaptation processes and outcomes. Of these, the major focus was on the means of measuring efficiency, effectiveness, usability of the systems and methods used, and subjective satisfaction.

There has been a trend towards the use of web-based tools for enabling dispersed groups of people to communicate and collaborate in performing their work [2, 5]. The use of such systems enables people to interact with each other easily by means of a web browser. The World Wide Web has become a familiar platform for constructing both Internet and organisational corporate intranets for use by many people. Some experimental studies have shown positive results for using web-based GDSS as a means of supporting communication in collaborative work – compared with traditional modes like face-to-face meetings [14]. Web-based GDSS, however, may lack security, have long periods of waiting for access time, and lack of integrity with standard DBMS. These disadvantages may be overcome when using the features of Lotus Notes®.

Because each participant working on a group task can act as an individual problem solver concentrating on specific aspects of the problem, asynchronous GDSS have to coordinate both individual and group problem solving processes. Different tools need to be used in different processes.

Whether the group support software should be integrated as a single system or use many different sub-systems to carry out different decision-making processes also has to be taken into account. Turoff, Hiltz, Bahgat, and Rana [25] suggested that the former is to be preferred because participants are unlikely to be satisfied with a system that requires different technologies and procedures when they are working as dispersed individuals contributing to a group, or when the group interacts simultaneously. If the sub-system
approach is chosen, it is necessary to ensure the consistency and integrity of data in different applications involved in the GDSS.

In recent years, two different approaches, decision conferencing and electronic meeting systems, have been proposed to support group work [17]. However, there are limitations on using these approaches. These limitations include the focus of the technology on information exchange rather than on decision making, and slow processing of text data by the group in the discussion phase. Pervan points out that intelligent decision support systems (IDSS) may be designed to tackle these limitations by enhancing and managing the processes, supporting specific applications and integrating those approaches to supporting group decision making. Perez, Garcia, Amescua, Ruiz, and Orga [15] also describe a generic DSS prototype, which has group support functions with Lotus Notes® and links with knowledge engineering modules implemented by Nexpert Object. Lotus Notes® provides the features that allow group participants to correspond with each other, whilst Nexpert Object loads the knowledge base and launches the alternatives for further evaluation.

There are several software applications, which support MCDM techniques under different environments. Most of them are based on the AHP method. For example, Expert Choice Pro, developed by Expert Choice Incorporation, lets users model a decision problem and evaluate the relative desirability of alternatives based on the AHP method [20, 7]. Expert Choice Pro supports such activities as brainstorming, problem structuring and criteria weighting [22]. Another MCDM system, AliahTHINK! has similar functions to Expert Choice but allows distributed users to open decision models separately via Lotus Notes® email infrastructure [1]. These systems, however, offer limited group support functions and only one decision model, i.e. AHP. In most cases they need group experts to provide their preferences one by one to the same model at different times. In this sense, they lack the support of some important GDSS features, e.g., parallelism and anonymity, especially in the idea-generation stage.

3. An intelligent MCDM process with a GDSS support

In general, MCDM process requires advance structuring to identify the problem, to explore the alternatives, and to develop and rank the criteria of the problem. This process can be done by a group of domain experts working together either through a series of traditional face-to-face meetings or with the support of a computerized GDSS. It seems impractical to bring groups of experts together for face-to-face meetings. Practically, every domain expert can contribute his or her ideas to the group at any place and any time with the support of an asynchronous GDSS.

A facilitator is responsible for controlling and monitoring the process through a facilitation component embedded in the system. He or she organizes the discussion and then aggregates individual preferences either by promoting consensus or using, for example, the geometric mean to summarize individuals' judgements [6]. Further below we describe the intelligent MCDM process assisted by an asynchronous GDSS.

Figure 1 presents such MCDM process with support of the asynchronous GDSS, which comprises an intelligent component.

![Figure 1. Intelligent MCDM process with the support of an asynchronous GDSS](image-url)

The intelligent MCDM process starts with the initiation of a group of experts chosen by the facilitator. The facilitator is responsible for manipulating and controlling the whole system. In the first stage of problem recognition, the facilitator receives the request from the user and organizes the group of experts who will participate in the problem structuring process. Occasionally, the facilitator will also be a member of the expert group.
Once the problem is structured and alternatives for the problem solving are identified, an intelligent shell comprising several MCDM models can be loaded. The intelligent component of the GDSS can guide in selecting a suitable MCDM model for different problems and tasks. The knowledge-base component of this system will help the facilitator to select an appropriate model for weighting criteria and evaluating the alternatives afterwards. Intelligent selection of the model depends on the type of task, the structure of the problem, the number of the criteria and alternatives, and so on. The GDSS support is available throughout the whole decision-making process.

There are two rounds of discussion in the decision-making process. In the first round, the experts are responsible for defining the problem, setting up the criteria and alternatives, and developing the structure of the problem. The individual opinions from the experts are stored in the discussion database and presented to the facilitator.

Two kinds of discussion procedures can be facilitated: consensus-seeking and aggregation. Consensus-seeking means that a final agreement among group members has to be reached, possibly by voting. Aggregation describes a procedure in which the experts enter their preferences and the facilitator summarises those preferences according to certain algorithms (e.g., the calculation of a geometric mean) to arrive at the final result. Because the input ('text data') from experts is organised structurally by the system, the facilitator can aggregate the input from the discussion database. With the assistance of the system he or she can filter redundancy (identical ideas resulting from parallel processes) first and then summarize the rest of it, aggregating similar ideas. This may require manual editing. The procedure tackles the problem of intelligent processing of text data suggested by Pervan [17].

The next step is for the facilitator to distribute the aggregated results to the experts. If a consensus needs to be reached, the experts may respond to the aggregated results by expressing their preferences again. This round of discussion continues until the problem is clearly structured. Then the facilitator calls on an intelligent (rule-based) component embedded in the system for the selection of an appropriate MCDM model according to the structure of the problem.

Once the MCDM model is selected, the second round of discussion begins. The expert group can be the same as the one in the first round of discussion, or different. The experts are asked to evaluate the criteria and alternatives from their points of view. Then the facilitator aggregates the individual preferences again and promotes the consensus (if necessary) through group correspondence. The result of the process will be that all the assessment data needed by the selected MCDM model are finally obtained.

The structure of the problem can be displayed diagrammatically as a hierarchy and attached with the experts' input documents (see Figure 2).

![Figure 2. Hierarchical structure of alternatives and respective criteria for car selection problem](image)

With the input of the available data into the selected MCDM model, the facilitator evaluates the alternatives by running the model. Final decision suggestions together with some appropriate explanations are either reported to the user or the group of experts for approval. The whole process is iterative (rather than strictly sequential) until the final results are generated by the system. All information obtained from this process is stored in the system database and can be retrieved for future decision situations.

4. Research methodology

4.1 The research question

The main purpose of our study is to ascertain the applicability of an asynchronous GDSS containing an intelligent component to guide users selecting a MCDM technique (model) in a Lotus Notes environment. It also aims to explore the effectiveness of
the system and satisfaction with its use in decision making by conducting a laboratory experiment and field test. The research question in our study is:

How can an asynchronous GDSS with an intelligent component developed in Lotus Notes® be used to improve a MCDM process?

The system (prototype) is being developed in a Lotus Notes® environment in order to take full advantage of its advanced features. The experiment is designed to investigate how use of the system might improve the performance of complex MCDM tasks.

4.2 Research model

Nunamaker, Dennis, Valacich, Vogel, and George [13] propose a GDSS research model which represents an input-process-output format. They argue that GDSS design is one of four contingencies (task structure, task support, process structure, and process support), along with the group, task, and context, that affects group processes and performance within these environments. This implies that GDSS research should not only explain why the outcomes occur but also deal with their application in real situations — the group, the task, the context, and the technology to which they apply (see Figure 3).

Most of the early GDSS research was in the form of laboratory experiments, but field research has increased in recent years. According to Gray et al. [9], laboratory experiments, case studies and field studies are the most prevalent research methods in GDSS. Konsynski and Stohr [10] compared laboratory experiments with field studies. They concluded that laboratory experiments have a high degree of internal validity (control) and low external validity (generalisability), whilst the reverse was true for field studies. Thus the trade-off between control and generalisability has to be considered. Pervan [16] advocates the conduct of more field research than laboratory experiments, because ‘the observation of using GDSS to solve real problems will provide greater opportunities for improving technological support for group decision-making than laboratory experiments with subjects’.

4.3 System (prototype) design

Our study follows a system development research methodology proposed by Nunamaker, Chen, and Purdin [12] – a recent version of which is presented in figure 4.
The system comprises four main components (see figure 5):
1. an asynchronous GDSS which can create, store, categorise, and retrieve relevant information for group discussion within a time constraint;
2. a facilitation support system which helps the facilitator assign the experts to the group, organise discussion, aggregate the data from the discussion database, and monitor the progress of the process;
3. a MCDM model base (‘intelligent shell’) which contains several available applications of MCDM techniques developed in different tools – e.g., LotusScript and C++; and
4. a rule-based system which guides users to select an appropriate MCDM technique (model) from MCDM model base.

**Figure 5. Architecture of the system**

A MCDM model base and a rule-based system interface to it comprise an ‘intelligent shell’ which contains a library of MCDM techniques with the ability to recommend the best for a particular decision situation. We call it a ‘shell’ in the same way as we talk about expert systems shells, which provide certain functionality and interface but need to be fed with some knowledge – i.e., in this context, a range of models.

All components of the system are built into an integrated messaging infrastructure of Lotus Notes®. This is necessary as Lotus Notes® offers a distributed client/server platform that allows applications and data to be shared by groups of users across a network. This infrastructure ensures that information is not only stored in or retrieved from the database between users and the system, but can be routed between users and even between different components of the system.

**4.3.1 Asynchronous GDSS component.** The asynchronous GDSS is a core component of the whole system. It contains a discussion database which stores structured information obtained from each expert of the group. The expert can respond either to the facilitator’s requests or other group member’s suggestions by entering his or her own preference for the problem, such as the definition, data hierarchy, and set of assessment criteria for the problem, the level of importance for each criteria, and so on.

This data is organised structurally in different fields by Notes™ Forms, so that relevant data can be retrieved afterwards. The predefined Notes™ Agents embedded in the facilitation support component ensure that all of the related information from the experts can be captured periodically and routed to the facilitator automatically for further aggregation. The discussion database holds all the information about the correspondence among the experts with the format of main documents, response to the main documents, and response-to-response documents. Information related to the discussion can thus be used for the future decision-making situations.

**4.3.2 Facilitation support component.** The facilitation support component assists the facilitator to organise, drive and monitor the current MCDM process efficiently. Detailed information about the available experts is recorded by name so that the facilitator can assign any of them to participate in the discussion. The facilitator is also responsible for maintaining security control, providing experts with various levels of access to the database.

The component shows the facilitator the current stage of the process data with Notes™. Input data from individual experts is sorted by the names of participants and the date it was created.

All the relevant data needed as an input to the intelligent (rule-based) component for the selection of a MCDM model can be identified afterwards either by the system (as far as possible automatically) or through dialog involving the facilitator. Notes™ Agents which have limited ‘intelligent’ features eliminate identical data, and the facilitator aggregates other similar ideas manually. The facilitator can also allocate access control passwords to the participants to ensure that data is only accessible to relevant participants. If necessary,
anonymity of the decision process can be also controlled by this component.

4.3.3 Intelligent MCDM shell.  
**MCDM model base component.** The model base component is a back-end of the intelligent shell. Our prototype contains several available MCDM techniques (models) including AHP, MFEP, and POUS (Lists Definition and Ordering Procedure, see [3]). The models in the model base are not restricted to the above three. Other models developed in C++ or LotusScript can also be included in the prototype afterwards so that a wider variety of problems and tasks can be dealt with in the future.

**Rule-based component.** The front end of the intelligent component in the system is a rule-based sub-system which is coded in LotusScript. It comprises rules that assist the user (facilitator) to choose an appropriate MCDM model from the model base. The rules are triggered according to the type of tasks, the definition and structure of the problem, the number of criteria and alternatives, and so on. This sub-system gets data which is aggregated by the facilitator and stored in the discussion database as input.

A selected model can be retrieved based on both the input data and additional interaction between the facilitator and the sub-system. The rules are flexible enough to be modified in case that more MCDM models are added in the future. There may be some situations when more than one rule can be applied. The explanations provided about each model should give enough information to the facilitator to make a final choice, or at least be aware of the limitations and advantages of using one model or another.

Lotus Notes® has a capability for integration with databases, which guarantees the longevity and integrity of the information. We are trying to build up the system with sole Lotus Notes® products such as Notes® Formulas, LotusScript and Lotus Components to ensure the ease of development and maintenance. Some procedures developed in other third-party packages may be integrated with Lotus Notes® through its Notes/FX feature. A data link interface (developed with Notes C++ API) which supports data transfer between different components (e.g., Lotus Notes® database and MCDM models developed in C++) is also built as an option. The data link interface will ensure that as much data as possible can be transferred automatically from Lotus Notes® discussion database into the MCDM model in order to relieve the burden of manual data entry by the facilitator.

4.3.4 Experiment design. In the past most of the experiments were conducted by comparing the effects of GDSS equipped groups with groups without a GDSS. Zigurs [26] criticizes that this two-way comparison is useful only when relatively simple systems are tested. He warns that if a system brings totally new paths to the decision process, then the supported group and non-supported group are incomparable. To address this problem, he advocates comparison of groups supported with different technologies.

We plan to set up the proposed system on two platforms: Lotus Notes® and the World Wide Web, since Internet has also been becoming one of the common IT infrastructure for the organizations. Lotus Notes® allows its database to be converted to Hypertext Markup Language (HTML) that can be accessed through the Internet with Web browsers [18]. We plan to organize the experiments with groups equipped with these two versions of the system to observe the different effects in terms of decision quality, user confidence, process quality, and commitment to results in given MCDM tasks. We will also observe the difference in terms of performance, cohesiveness and general satisfaction among different groups.

5. Conclusion and future research

There are three distinctive phases in our research project:

1. Development of the prototype;
2. Laboratory experiment;
3. Field test.

The system (prototype) design has been currently finished and the prototype implementation is in progress. Once the prototype is built, experiments will be organized in order to observe its impact on the process of group decision-making. Both laboratory experiment and field test will be conducted at this phase of the study.

Our study aims to build an asynchronous GDSS with an intelligent component which supports MCDM processes in Lotus Notes®. It also aims to find a way to improve the MCDM process comprehensively. The prototype itself can be used to support communication
at any-time and any-place but also to select a MCDM model.

Effective use of the prototype will enable practitioners to improve the MCDM process, and make for a better understanding of the use of GDSS. It might not be possible to generalise from this study across the whole spectrum of GDSS use, but it can provide insights into relevant fields. Therefore, this research has the potential to make a contribution to the existing theory and the use of both asynchronous GDSS and intelligent MCDM processes.

6. References


