Improving User and Decision Support System Teamwork

An Approach Based on Shared Mental Models

Iris van de Kieft, Catholijn M. Jonker, and M. Birna van Riemsdijk

Man Machine Interaction Group, Delft University of Technology, The Netherlands

Abstract. Social psychology research on teams has shown that team performance improves when team members have a so-called shared mental model (SMM). We observe that a decision support system (DSS) and its user can be viewed as a team whose task is to make a (good) decision. Therefore we argue that it is important for good user-DSS team performance that they have a SMM. In this paper we make precise what it means for a user and DSS to have a SMM. This can be used as the basis for analyzing to what extent a DSS supports the achievement of a SMM, or to design a DSS such that it does this. We apply our framework to a particular negotiation support system, using it to analyze to what extent this DSS facilitates the achievement of a SMM. We propose that the addition of explanation facilities to the DSS can improve sharedness in the user-DSS team.

1 Introduction

Decision support systems (DSSs) are interactive computer systems that aid users in judgement and choice activities [5]. The user and the DSS cooperate, to structure the decision problem and make a choice. We observe that the user and the DSS can be viewed as a *team* whose goal is to make a (good) decision.

It is known from social psychology research that team performance improves if team members have a shared understanding of the task that is to be performed and of the involved teamwork, i.e., if team members have a so-called *shared mental model* (SMM). SMMs are defined in [3] as:

> knowledge structures held by members of a team that enable them to form accurate explanations and expectations for the task, and, in turn, coordinate their actions and adapt their behavior to demands of the task and other team members.

We argue that research on SMMs is not only relevant for human teams, but also for human-DSS teams. In other words, we expect that improving user-DSS teams' SMM will improve their team performance. User-DSS teams form a specific kind of team. Most notably, a user-DSS team consists of a human team member and a software agent, while human-only teams have been the focus of SMM research. This also results in each team member having its own, complementary capabilities. For example, compared to the DSS, the user possesses more general world knowledge, whereas, compared to the user, the DSS has superior computational abilities. Moreover, it is a hierarchical team in which the DSS supports the user. The user makes the final decision, for which she is responsible[1].

In this paper we make precise what it means for a user and DSS to have a SMM. This can be used as the basis for analyzing to what extent a DSS supports the achievement of a SMM, or to design a DSS such that it does this. To be more specific, we analyze the user-DSS task and team, based on existing categories of mental models from SMM theory [3] (Section 2). This allows us to determine what knowledge is relevant for this task and team. We use this analysis to make precise what a SMM is in the context of user-DSS teams, based on the identification of two types of sharedness that we consider relevant in user-DSS teams (Section 3). We discuss how sharedness can be achieved and maintained in Section 4. In Section 5 we apply our framework to a particular negotiation support system by using it to analyze to what extent this DSS facilitates the achievement of a SMM. We propose that the addition of explanation facilities to the DSS can improve sharedness in the user-DSS team, where our framework helps determine the requirements for such explanations. We conclude the paper in Section 6.

2 Analysis of User-DSS Task and Team

In this section we analyse user-DSS teamwork, based on the mental model categories defined in [3]: task, equipment, team members[2] and team interaction. For each mental model, we discuss its contents, given the user-DSS teamwork context.

2.1 Task

The knowledge content of the task model concerns task procedures, strategies, likely contingencies, and environmental conditions [3]. In user-DSS teamwork, the task is *decision making*, and thus involves decision making knowledge. H.A. Simon identified three phases in the decision making process: intelligence, design and choice [16]. In the intelligence phase, the problem or opportunity is identified. In this phase, the decision problem and its domain are defined.

In the design phase, a decision model is constructed. This involves identifying the important criteria for the decision process and for the decision result, and constructing the available options. A decision model and its variables represent preferences over decision criteria, available options and a measure of uncertainty over the variables influencing decision process and result [5]. We note that the preferences in the decision model are often those of the user, as she is the decision maker. However, the preferences may also be determined by external parties. For example, in managerial decisions, the preferences may be determined by the user's superiors.

In the choice phase, the available options are evaluated, taking into account preferences over decision criteria and measures of uncertainty, after which the best decision option is selected. In this phase a strategy or choice method is needed, to make

[1] Here, we assume the user always remains in control of the final decision. We do not discuss cases of adjustable autonomy [15].

[2] We refer to the team mental model as the team member mental model, to avoid confusion with the concept of team mental model as used in, for example: [12].

a choice amongst the available options. Thus, the knowledge about 'task procedures' and 'strategies' consists of the above described decision making knowledge. 'Likely contingencies' are taken into account by the measures of uncertainty mentioned above. Furthermore, we take 'environmental conditions' to mean the relevant knowledge about the current state of the environment, as well as relevant previous states.

2.2 Equipment

The knowledge content of this model concerns equipment functioning, operating procedures, system limitations, and likely failures [3]. For any equipment that the user-DSS team makes use of, these four knowledge elements are needed. Although the DSS can be viewed as technology or equipment, we view it as a team member, and thus do not consider knowledge about the DSS to be part of this category.

2.3 Team Members

The knowledge content of this mental model concerns knowledge, skills, strengths, limitations, preferences and tendencies of teammates [3]. We combine the 'skills' and 'strengths' elements, as for our purposes, these are quite similar. Additionally, for now we leave out the 'tendencies', as this is a more subtle aspect that is relevant for human-human interaction, whereas for human-agent interaction it is less clear how this applies. In the following we discuss these knowledge content elements in the context of user-DSS teams.

Knowledge: The user and the DSS typically possess partially distinct knowledge. For example, the user has a wealth of knowledge about world, about interacting with other humans and about her own preferences. The user likely possesses certain decision making knowledge such as knowledge about the domain and the available options. The DSS has specific decision making knowledge about how to structure the decision problem, and has one or more choice methods.

Skills and strengths: The user and the DSS have different, complementary skills. For example, the user can interact easily with the world, and gather external knowledge about current state of the environment. The user has emotions that in some cases can help decision making [4]. The DSS is a specialist in the decision making task it was built for. It has good memory, superior computational abilities and remains rational.

Limitations: The user need not be a specialist in the task, has emotions and thus may not always be rational, has limited (working) memory and limited computational power. The reasoning of the DSS is limited to the task it is built for. It may not be able to adapt to new domains or contingencies. The user and the DSS thus, to a large extent, have complementary skills and limitations.

Preferences: The user has preferences that may be key to the decision making process. In this case, they should be part of the task mental model. The user may also have other preferences that relate more to the interaction style she prefers, in other words, preferences related to team interaction, not task. These should then be part of the team member mental model. The DSS does not have preferences.

2.4 Team interaction

The knowledge content of this model concerns roles/responsibilities, communication channels, interaction patterns, information flow, role interdependencies and information sources [3]. For each element of the knowledge content, we discuss what its contents are in the context of user-DSS teams.

Roles/responsibilities: Based on their complementary capabilities, each team member has a specific role and corresponding responsibilities. The role of the user is to provide the DSS with the required information, and other tasks such as interaction with other humans. The user is responsible for making the final decision and for its consequences. The role of the DSS is to assist in structuring the decision, provide choice advice, request necessary information from the user and help the user understand the process and construct domain and preferences.

Communication channel: An important characteristic of user-DSS teams is that the interaction takes place via a user interface. Thus, the set of interactions is limited to those that are allowed by the user interface. This sets this type of teamwork apart from human teamwork, where interactions are not necessarily as constrained.

Interaction patterns and information flow: The interaction between user and DSS starts when the user has a decision to make, and requests assistance from the DSS. Together, the DSS and the user identify the decision problem and its domain (intelligence phase), construct the decision model (design phase) and make a choice (choice phase).

In the intelligence phase, the user informs the DSS of relevant domain and decision knowledge she has access to. The DSS provides information to the user about structuring the decision problem, about the domain and decision knowledge it has, why and when it requires information from the user, extra information that helps the user understand the decision problem, etc.

In the design phase, the user provides information on, for example, preferences, environmental conditions and available options. The DSS provides information about the decision model it constructs, and again, any additional information that helps the user understand the decision problem and the DSS functioning and interaction.

In the choice phase, we come across an interesting characteristic of this type of decision making, due to the hierarchical nature of the team. Because the DSS has superior computational abilities, it can often more easily evaluate the options and choose what it believes to be the best one. However, because the user remains responsible for the decision making, the DSS ‘decision’ is merely advice that is presented to the user. The user must then evaluate this advice, and make the final decision to either accept the advice, or choose another option.

Role interdependencies: The user relies on the DSS to help her structure the decision and provide advice, in other words, to take over some of the reasoning. The DSS usually requires user input on, for example, domain, preferences, environmental conditions. It also relies on the user to make the final decision.

Information Sources: The user and the DSS may use additional information sources. For example, some DSSs assist decision making by integrating various sources of information such as external databases.

3 Sharedness in User-DSS Teams

In this section, we use the analysis of user-DSS teamwork as performed in the previous section, to make precise what a SMM is in the context of user-DSS teams. This can be used as the basis for analyzing to what extent a DSS supports the achievement of a SMM, or to design a DSS such that it does this.

First we identify two types of sharedness, based on the observation that DSSs often do not have explicit mental models about certain aspects of the teamwork (Section 3.1). Second, we discuss for each mental model category which type of sharedness applies and which knowledge should be shared and what can remain distributed (Section 3.2).

3.1 Types of Sharedness

As mentioned previously, we expect appropriate shared knowledge between user and DSS to improve team performance. However, as different definitions of 'shared' are possible, it is important to define what is meant by this term [2]. For user-DSS teams, we distinguish two types of sharedness.

In Type 1, the mental models of team members are compared directly and sharedness is defined as the extent to which these mental models overlap (see Fig. 1: Arrow 1). Type 2 stems from the observation that in user-DSS teams, we cannot always directly compare the mental models of these team members, because the DSS does not always have explicit mental models. For example, the DSS does not always have a mental model of itself and the interaction; in this case, the relevant knowledge is implicitly present in the DSS design. In such situations, we compare the mental model of the user directly to the DSS design (see Fig. 1: Arrow 2). This comparison determines if the user's mental model is an accurate (or at least, compatible) representation of the DSS.

These two ways of comparing mental models are similar to the approach taken in [1], where a distinction is made between comparing mental models directly, and comparing a mental model of one agent with the beliefs another agent has about this mental model.

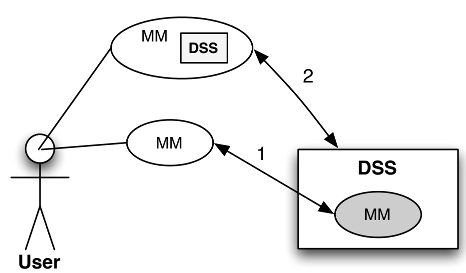

Fig. 1. Two types of sharedness. Type 1 involves comparing the user's mental model to the DSS's mental model. Type 2 involves comparing the mental model the user has of the DSS (or the user-DSS interaction), to the DSS itself.

3.2 Desired Shared Mental Model

It cannot be expected that user and DSS have identical knowledge, nor is it desirable, since user and DSS have distinct roles and distinct skills (see Section 2.3). Within the context of heterogeneous teams with distinct roles, of which user-DSS teams are a good example, too much overlap in knowledge can result in redundancy and inefficiency, and contribute to sub-optimal use of resources [14]. Nevertheless, because of their role interdependencies, some knowledge must be shared, to ensure that the user and the DSS have a general shared understanding about the task.

The idea that some, but not all, knowledge should be shared between user and system, is in line with the findings of a study on explanation in knowledge based systems (KBSs) [13]. The authors found that users requested explanation only when deemed necessary and essential for task completion. Such explanation requests were triggered by perceived anomalies in KBS output. In other words, users usually only requested information when they did not understand KBS output that was important for task completion.

We will now discuss, for each mental model category, which type of sharedness applies, and what knowledge should be shared and what can remain distributed. We call the knowledge of a mental model that should be shared the *desired SMM*.

Task mental model: Both user and DSS have a mental model of the task, thus, sharedness Type 1 applies here. Which knowledge elements of the task mental model should be shared or remain distributed depends largely on the specific domain and context.

In general, the DSS will have more detailed knowledge about the task, as it is built specifically for it. However, because the user is responsible for some of the DSS input, and for the final decision, some general task knowledge should be shared. For example, given the correct input, the DSS can calculate the entire option space. The user often cannot, and need not, keep track of all the available options. Nevertheless, some of this knowledge should be shared with the user, as the user will have difficulty making the final decision without some idea of the available options.

Equipment mental model: The user and the DSS should have a mental model of any equipment that is relevant for their part of the task. Thus, here, mental models can be compared directly, and therefore, the sharedness for this mental model category is of Type 1. For this mental model, we also see that detailed knowledge often resides with one team member, but general knowledge is shared. That is, the user and DSS each need a detailed mental model of the equipment that they themselves use. By contrast, if a team member does not operate certain equipment, but its results are relevant for the task, (only) some general knowledge about the equipment should be shared.

Team member mental model: As there are two team members, we consider two different mental models, one concerning the DSS and one concerning the user. First, we consider the team member mental model concerning the DSS. We observe that the user should have such a mental model. However, as most DSSs are not able to explicitly reason about their own knowledge and abilities, the DSS usually does not have an explicit model of itself. Thus, Type 2 sharedness applies here: the mental model that the user has of the DSS is compared directly to the DSS (design). The user's mental model of the DSS need not contain every detail about the DSS. Rather, it should contain sufficient information about DSS knowledge, skills/strengths, limitations, so that team performance can be successful. This is thus the desired SMM.

Second, we consider the team member mental model concerning the user. We assume that the user has a mental model of herself, in so far that she is explicitly aware of her own knowledge, skills, limitations and preferences. If the DSS does not have an explicit user model, there is Type 2 sharedness: the mental model of the user about herself can be compared to the user characteristics as integrated into the DSS design. That is, the DSS is typically built with certain intended users in mind. Certain characteristics of these intended users are integrated into DSS design. For example, a medical DSS intended for doctors will be built differently from a medical DSS that has laymen as intended users. We note that in this case the level of sharedness cannot be changed, as the intended user characteristics built into the DSS cannot be changed, nor can the user adapt her mental model about herself to fit these intended user characteristics.

If the DSS does have an explicit user model, Type 1 sharedness applies, where the DSS's mental model of the current user is compared to the mental model the user has of herself. The former can be used to create more sharedness between the DSS and the user regarding this team member mental model, as knowledge about current user characteristics allows the DSS to *adapt* the interaction to this user (see e.g., [8]). . For the team member mental model concerning the user, the desired SMM contains the user characteristics (knowledge, skills, limitations, preferences) relevant for adaptation. For example, if the DSS should be able to adapt to the user's expertise level, it should have knowledge about this expertise level. Any other user characteristics remain distributed, residing with the user.

Team interaction mental model: The user has a mental model of the interaction, whereas the DSS usually does not have such an explicit mental model. Thus, also here, Type 2 sharedness applies. The user's mental model of team interaction is compared with the team interaction as integrated into DSS design. Again, the user needs sufficient team interaction information to achieve successful interaction. This is the desired SMM.

4 Achieving Sharedness

The SMM framework presented in the previous sections provides a basis for determining what knowledge should be shared between user and DSS. The next step is to address how such sharedness can be achieved and maintained. We define a *discrepancy* to be a lack of sharedness regarding a knowledge element. As discrepancies may also exist for elements that need not be shared, we will sometimes use the term 'problematic discrepancy' to emphasize that we are referring to a discrepancy regarding an element of the desired SMM.

To achieve the desired SMM, problematic discrepancies should be resolved. First the discrepancy must be detected, then it can be resolved by adapting (one of) the mental models. We argue that explanation capabilities on the part of the DSS are important for achieving sharedness in mental models. That is, a DSS should be able to explain to the human user its reasoning and the input upon which it bases its reasoning. Explanation has been studied in the context of expert systems [11] and is increasingly being investigated in a wide variety of intelligent systems such as intelligent virtual agents [6] and recommender systems [17].

Explanation can serve various purposes, such as improving effectiveness, increasing the user's trust in the system and improving transparency of the system [17]. We are particularly interested in the latter, as this facilitates detecting and resolving discrepancies between mental models of DSS and user. Transparency means explaining how the system works, thus giving the user a better understanding of the DSS's reasoning process. This allows the user to detect any discrepancies between the mental models, and subsequently to resolve these by updating the mental models where necessary. Our framework can form a basis for these explanations, determining what should be shared, and thus what should be explained. That is, when adding an explanation facility to a DSS, our framework can be used to determine the requirements for the explanations. Such an explanation facility should provide the transparency necessary to detect and resolve problematic discrepancies. Hence, the explanations should be targeted at the content of the desired SMM.

5 Case Study: Negotiation Support System

In this section we apply our SMM framework to a concrete DSS: the Pocket Negotiator [7]. The Pocket Negotiator is a negotiation support system (NSS). Its aim is to provide focus and structured support for the negotiation process, increasing the user's capacity for structuring and exploring the negotiation space, and reducing the cognitive task load while doing so. The Pocket Negotiator currently exists as a web application protoype (demo video: `http://mmi.tudelft.nl/?q=node/40`). This prototype guides the user through the four major stages that can be discerned in integrative negotiation: private preparation, joint exploration, bidding, and closing. The actual decision making occurs during the bidding stage, when the user must make a decision about about what bid to place next, or whether to accept the opponent's bid.

Using our framework, we provide some examples that illustrate the extent to which this DSS supports the achievement of a SMM, and where there is room for improvement. These examples also illustrate how explanation can improve sharedness.

Task mental model: As a first example, we look at *user preferences*. In the Pocket Negotiator, user preferences are elicited and visible in the preparation and exploration phase. This part of the interface has been designed to help the user explicate her preferences, and to allow sharing this information with the NSS, thereby facilitating achievement of sharedness.

However, during the bidding phase, the user preferences are only implicitly visible in an overall utility rating per bid. The Pocket Negotiator uses the user preferences to determine a utility function, which assigns a utility value to each available option. The link between the user preferences and the assigned bid utility is not very clear, making it difficult for the user to monitor if the DSS has a correct mental model of user preferences and the corresponding utility function. This could be improved using explanation, to maintain sharedness regarding preferences and utility function (cf. [10]). Sharedness regarding user preferences is important, because these preferences form the basis for the choice strategy. If the DSS has incorrect information regarding the user's preferences (a problematic discrepancy), it will not be able to provide appropriate advice.

An example of a knowledge element for which sharedness is clearly not facilitated in the Pocket Negotiator, is the *strategy*. The Pocket Negotiator does not make clear to the user what kind of strategy it is using. Also, the user has no option to choose a strategy, and thus cannot achieve sharedness in this manner. This lack of sharedness regarding strategy makes it difficult for the user to understand the Pocket Negotiator's advice, giving rise to a problematic discrepancy. If the Pocket Negotiator were to explain how it used the decision model and choice strategy to come to its advice, it would be easier for the user to evaluate the advice. Here of course, the Pocket Negotiator need not share all the details of the choice strategy it uses. The choice strategy involves a number of calculations, which the user does not need to understand fully. It should be possible to provide a reasonable explanation to the user, without going in to the details of the calculations.

Team member mental model concerning user: The Pocket Negotiator design has taken into account the knowledge, abilities and limitations of its intended users. For example, the preference elicitation interface takes into account that users often find it easier to think about their interests than their specific preferences per issue. However, the Pocket Negotiator does not store any knowledge about the current user that it could use for adaptivity purposes. If the Pocket Negotiator were able to have a mental model of the user's level of expertise, and adapt accordingly, this could greatly improve the team performance. For example, novice users may need more explanation to gain sufficient understanding of the decision problem, whereas expert users might get frustrated by too much explanation. Indeed, research has found that novices and experts have different ways of making use of explanations [13].

6 Conclusion

In this paper we have made precise what it means for a user and a DSS to have a SMM. We analyzed the task and team, to specify the relevant knowledge. Subsequently, we specified what we mean by 'shared', defining two types of sharedness that are relevant for user-DSS teams. We then proceeded to describe what knowledge should be shared between user and DSS, and what knowledge can remain distributed. A case study of an NSS illustrated our approach, where we identified explanation as suitable for improving sharedness. Our work can be used as the basis for analyzing to what extent a DSS supports the achievement of a SMM, or to design a DSS such that it does this.

In future work we plan to perform a more elaborate case study. This should include user studies to determine the effect of (different kinds of) explanation on sharedness, and to determine if improved sharedness indeed improves user-DSS team performance. In order to perform such user studies, we must first define ways in which to measure sharedness beween user and DSS, to be developed on the basis of [9].

Acknowledgements This research is supported by the Dutch Technology Foundation STW, applied science division of NWO and the Technology Program of the Ministry of Economic Affairs. It is part of the Pocket Negotiator project with grant number VICI-project 08075.

References

1. Burgemeestre, C.B., Liu, J., Hulstijn, J., Tan, Y.: Early requirements engineering for e-customs decision support: Assessing overlap in mental models. In: Proceedings of the CAiSE Forum. pp. 31–36 (2009)
2. Cannon-Bowers, J.A., Salas, E.: Reflections on shared cognition. Journal of Organizational Behavior 22(2), 195–202 (2001)
3. Cannon-Bowers, J.A., Salas, E., Converse, S.A.: Shared mental models in expert decision making teams. Current Issues in Individual and Group Decision Making pp. 221–246 (1993)
4. Damasio, A.R.: Descartes' Error: Emotion, Reason, and the Human Brain. Putnam Publishing (1994)
5. Druzdzel, M.J., Flynn, R.R.: Decision support systems. In: Kent, A. (ed.) Encyclopedia of Library and Information Science. New York: Marcel Dekker, 2nd edn. (2003)
6. Harbers, M., van den Bosch, K., Meyer, J.J.: A methodology for developing self-explaining agents for virtual training. In: Proceedings of the 8th International Conference on Autonomous Agents and Multiagent Systems: Volume 2. pp. 1129–1130 (2009)
7. Hindriks, K.V., Jonker, C.M.: Creating human-machine synergy in negotiation support systems: Towards the pocket negotiator. In: Proceedings of the 1st International Working Conference on Human Factors and Computational Models in Negotiation. pp. 47–54 (2008)
8. Hudlicka, E., McNeese, M.D.: Assessment of user affective and belief states for interface adaptation: Application to an air force pilot task. User Modeling and User-Adapted Interaction 12(1), 1–47 (2002)
9. Jonker, C.M., van Riemsdijk, M.B., Vermeulen, B.: Shared mental models: A Conceptual Analysis. In: Coordination, Organization, Institutions and Norms in Multi-Agent Systems at AAMAS2010 (2010)
10. van de Kieft, I., Jonker, C.M., van Riemsdijk, M.B.: Explaining negotiation: Obtaining a shared mental model of preferences. In: IEA/AIE 2011, Part II, LNAI 6704. pp. 120–129. Springer Verlag (2011)
11. Lacave, C., Diez, F.J.: A review of explanation methods for heuristic expert systems. The Knowledge Engineering Review 19(02), 133–146 (2005)
12. Langan-Fox, J.: Skill acquisition and the development of a team mental model: an integrative approach to analysing organizational teams, task, and context. In: West, M.A., Tjosvold, D., Smith, K.G. (eds.) International Handbook of Organizational Teamwork and Cooperative Working, chap. 16, pp. 321 –360. John Wiley & Sons (2003)
13. Mao, J., Benbasat, I.: The use of explanations in knowledge-based systems: Cognitive perspectives and a process-tracing analysis. Journal of Management Information Systems 17(2), 153–179 (2000)
14. Mohammed, S., Dumville, B.C.: Team mental models in a team knowledge framework: Expanding theory and measurement across disciplinary boundaries. Journal of Organizational Behavior 22(2), 89–106 (2001)
15. Scerri, P., Pynadath, D.V., Tambe, M.: Why the elf acted autonomously: Towards a theory of adjustable autonomy. In: Proceedings of the 1st International Joint Conference on Autonomous Agents and Multiagent Aystems: part 2. pp. 857–864. ACM (2002)
16. Simon, H.A.: The new science of management decision. Prentice-Hall, Englewood Cliffs, N.J. (1960)
17. Tintarev, N., Masthoff, J.: A survey of explanations in recommender systems. In: ICDE'07 Workshop on Recommender Systems and Intelligent User Interfaces. pp. 801–810 (2007)