Control and Cybernetics

vol. **50** (2021) No. 1

E-negotiation design and engineering*

by

Gregory (Grzegorz) Kersten

Abstract: This paper is an excerpt from a book by Gregory Kersten, which was not yet completed before his unexpected passing in May 2020. Following a note from Gregory's family, the paper includes a section of a book chapter dealing with negotiation protocols. It shows how negotiation, as a purposeful process governed by explicit and implicit rules has protocols that can be employed to help negotiators orient themselves in this process. In particular, with respect to e-negotiations, the paper stresses the need to have a shared and clear understanding of the terms that are used by negotiators and software, which should be provided by taxonomy or ontology. The paper distinguishes three principal aspects of negotiations: decision and choice; language; and process. It also identifies key challenges related to the successful implementation of negotiation software agents.

Keywords: e-negotiation protocols, e-negotiation support systems, negotiation ontology, software agents

A note from Gregory Kersten's family

In the early 2000s, Gregory Kersten started to write a book on negotiations with the working title of *Negotiations and E-negotiations: Analysis, Management and Support*. Over the years, the scope of the book grew, alongside with its length and depth. Parts of it weaved their way in and out of the many papers that Gregory wrote, for which he would often enlist us, his wife and children, to proof read and copy-edit. The topics he covered surfaced during much-cherished dinners, family travels, chairlift rides, and one-on-one conversations.

Gregory's unexpected passing in May 2020 left his book unfinished. We are considering what to do with this work. In the meantime, we wanted to share an excerpt of the book with the *C&C* journal, as this community held an important place in Gregory's heart and mind.

The text provided here has been left un-edited, with the exception of footnotes, citations, and minor grammatical changes*. What follows is the intro-

*Submitted and accepted in December 2020.

*some insignificant changes have been introduced due to adjustment to the journal style and rules (eds.).

duction to the book as well as part of Chapter 18, entitled *E-negotiation design and engineering.* The excerpt here provided is section five of the Chapter, which is entitled *Negotiation protocols.* It focuses on taxonomy, ontology, and protocols. It describes the characteristics of negotiation protocols, sequences and processes, their rules and the transition between them.

The full Chapter consists of six sections. The first section, *Internet and negotiations*, focuses on e-negotiation systems, including socio-technical systems and their functions. Section two, *Software engineering,* focuses on domain engineering, requirement engineering, and software design. Section three, *The extended Montreal taxonomy* describes media reference and negotiation phase models, taxonomy and top-level structure, negotiation constructs, and problem types. Section four is devoted to mechanism design as well as an examination of different mechanisms, including market mechanisms. Section five is reproduced below. The concluding section of the chapter, *Protocol representation* deals with sequences, processes and rules, protocol modification as well as closed protocols.

Gregory cared deeply about those studying and pushing the boundaries of research on negotiation and e-negotiation. We hope that this excerpt sparks interest in those passionate about this topic.

If you are interested in receiving access to the other sections of the chapter or learning more about Gregory Kersten's manuscript, please email negotiation-book@kerstens.org.

Margaret, Mik, Marta, and Mark Kersten

Book Introduction

There is a long tradition of studying conflict and negotiations from such perspectives as psychology, sociology, political science and management science. Negotiations today, as so many other human endeavours, involve information and communication technologies. Some negotiations are conducted via email, fax, and videoconferencing systems. Other negotiations involve the use of software packages to determine options, analyze scenarios, and conduct simulations. Yet other negotiations involve software agents that automate certain tasks or even the whole process. In effect, research in negotiations and in tools for negotiations has been undertaken in computer science, software engineering and management information systems.

In the past, there was often little relationship between human and social sciences on one hand, and engineering and computer science on the other. Behavioural researchers used tools and software; engineers and computer scientists implemented models and rules resulting from laboratory and field experiments.

Increasingly, the designers and developers need to have comprehensive knowledge about future users of their products. Their products affect the users' faculties, the way they perceive, reason and communicate; they also change the social groups and socio-economic interactions.

One purpose of this book is to provide software designers and developers with a single point of entry where they can find knowledge about negotiators and negotiations coming from human and social sciences. Therefore, several chapters are devoted to behavioural issues with an effort to present this knowledge in a structured and organized manner. To this end a common terminology is proposed, representations providing high-level perspectives are formulated, and theories and models are categorized.

In a similar manner the book presents formal models and procedures. Many of them have already found their ways in software; others may be used in the future. They may also provide a basis to design new models and procedures. These models and procedures also provide a bridge between behavioural research and engineering. They help to gain insights into standard types of behaviours and understand the opportunities for activities undertaken by software.

The software, its use and potential are discussed in detail. The purpose is to provide human and social scientists with a single point of entry to the current situation in the use and development of e-negotiation systems and tools. Inasmuch as engineers and computer scientists need to gain a deep understanding of the human and social perspectives and ways they may impact these perspectives, behavioural researchers need to understand the power of software in performing cognitively complex tasks and replacing humans in many mundane and difficult activities.

Negotiation is a fascinating and multifaceted process. It is not possible to present in any detail all of its facets and provide all the linkages that contribute to its richness and complexity; the more so that there is no one negotiation type or category. Negotiation is a process of managing and resolving conflicts, hence it is a positivistic process. In a similar manner, putting together the many different approaches, perspectives and models of negotiators and negotiations has a positivistic aim. This aim is to aid researchers, designers and developers in their work and provide students with a source, which they can use to better understand the many aspects of negotiations.

1. Negotiation protocols

† In some situations, we may design a single negotiation mechanism and implement it in software. Consider the web-assisted claim settlement mechanism of Cybersettle‡. It is simple in that it requires both parties to enter three offers and splits the difference between two overlapping offers. If there are no overlapping offers, then the parties may use a human facilitator; terminating the use of software.

The Cybersettle settlement mechanism is simple, but it also requires other mechanisms to be implemented so that its users can be assured of security,

†as mentioned before, the text provided is Section 5 of Chapter 18 of the book (eds.).

‡http://www.cybersettle.com

privacy, and non-reputation. The software may need mechanisms to maintain its reliability, achieve ease of use and provide linkages with telephone services. Thus, even if a single mechanism is at the core of the ENS (electronic negotiation system), typically there are other mechanisms implemented in the system.

There are complex negotiations, in which the participants may wish to cooperate with agents (human and/or software), use tools for risk assessment and preference elicitation, and use visualization tools, etc. These ENSs will have many mechanisms and the question is about their use for a particular negotiation. Protocols, which in negotiations between the representatives of governments have been used to structure the process and guide the parties, can provide similar function in e-negotiations. They can organize the process and coordinate the use of different mechanisms.

The distinction between mechanisms and protocols is not strict; it may be possible to construct a single mechanism that controls all activities leading to the achievement of a particular goal. Because such a mechanism would have components coming from different domains, and the relationship between components would be complex, an internal organizing mechanism may be necessary. Therefore, it is often easier and simpler to distinguish between protocols and mechanisms. A simple rule may be that mechanisms tell their users what they can do to use the mechanism, while protocols tell them when they can use a mechanism and what other (possible alternatives) mechanisms they can or will use.

1.1. Taxonomy, ontology, and protocols

Protocols help the negotiators to orient themselves in the process. They also organize software components that participate in the process. Therefore, they need to use terms that can be unequivocally understood by the negotiators and software. Because of a plethora of overlapping and contradicting terms, it is necessary to use a taxonomy or ontology, if one exists.[§]

1.1.1. Negotiation constructs and ENS components

Negotiation is a purposeful process governed by explicit and implicit rules. In face-to-face negotiations many of the rules are ill-defined and some are clarified only during the process. Some rules may be explicit while others are implicit. The rules may be rooted in the parties' cultural and professional backgrounds and applied in every given situation. Certain rules of communication and offer exchange may be established during the initial phase of negotiations. Other

[§]These two terms, taxonomy and ontology are sometimes used interchangeably and sometimes are assigned different meanings. Taxonomy is a hierarchically organized list of terms pertaining to a domain of interest. Ontology is a taxonomy that also contains grammar that can be used to associate terms to express the behaviors, properties and relationships in the domain.

rules may be established ad hoc, for example, when the parties are close to an agreement, they may decide on the "split-the difference" rule.

Behavioral researchers proposed rules and other models for different negotiation types and negotiators' characteristics and behaviors. These rules are based on the studies of real-life and simulated negotiations. Their purpose is to provide prescriptions for students and practitioners.

Rules and other models constructed with the purpose of advising negotiators may be ambiguous, due to the assumption that the negotiators know when and how to use them. This lack of precision gives the negotiators flexibility in adapting the rules to a particular situation.

In face-to-face negotiations, many of the constructs comprising the extended Montreal taxonomy (EMT) are rarely fully specified. An attempt to specify many constructs would impose prohibitive cognitive, information and time requirements on the participants. The effort and time required for collecting and processing information about the various constructs would lead to learning and obtaining a thorough understanding of the participants, the problem and the negotiation process at the cost of finding an agreement efficiently. In the extreme, the negotiators would spend so much time on gathering and verifying information that they would have no time to negotiate.

E-negotiations require that some constructs be clearly specified. What and how many constructs are fully specified depends on the roles and functions of the ENS. In automated negotiations the construct specification needs to be *complete* so that the software agents can undertake the required negotiation tasks and be able to reach an agreement. In e-negotiations that involve people, some constructs are embedded in the system and some are left for the people to formulate. In email-based negotiations the key specification is message attributes (e.g., recipient address, time stamp, and message title). Other specifications may refer to message threading and a deleting policy. There are not many constructs that need to be defined in email-based negotiations, because software activities are limited to communication, storage and display of information; software does not participate in the constructs' formulation and evaluation.

In most ENSs the constructs' specification is *incomplete*, with some constructs being completely implemented in software, some partially implemented, and the implementation of some being left to the users.

Negotiation constructs are implemented in the ENS components. The execution of a software component transforms information obtained from the ENS users, other components, and external sources into actions and updates constructs.

The selection and implementation of the negotiation constructs, and the information exchange are important ENS design activities. Specifically, the *main ENS design issues* are:

1. The specification of the constructs that are formulated solely by the system, those that are formulated jointly by the system and its users, and those that are the sole responsibility of the users;
2. The selection of models and procedures used for construct implementation;
3. The design of ENS and its components together with the assignment of constructs to components; and
4. The specification of the communication among components, between the components and the ENS users, and with other systems.

During the design phase of an ENS, decisions are made regarding the division of work between the system and its users. When they partake in the e-negotiation, they collect, process and produce information in order to achieve outcomes that the users desire. The ENS activities have to follow rules and procedures, which must be well defined. The users also need to follow certain rules to provide the system with information and obtained required output. The collection of these rules and procedures is the negotiation protocol.

1.1.2. Decision, communication and process

Negotiation constructs can be used to describe the negotiation and its structure. They also help to specify the permissible negotiators' behaviors and conditions for their movement through the process; such a description is known as a negotiator protocol (see Ströbel, 2001; De Dreu, Weingard and Kwon, 2000). This view of the protocol deals with different communication acts but not with their content. It restricts the participants' moves but gives them the freedom to do anything they wish when they are in a given state.

Software agents do not yet have the degree of intelligence and common sense that would allow them to function effectively when their communication content and form are not prescribed. In automated negotiations and also when software agents' aid people, the content of the agents' communiqués is determined by a protocol (Muller, 1996). It is not only software agents that may need such a protocol; human negotiators may also need help in making sure that they communicate using language and terms that convey the intended message and in a form that is acceptable to the recipient (e.g., is polite and dutiful).

Protocols may also guide the actions of both human and software agents' independent activities, such as preference assessment, search for a counterpart, and offer analysis decision. Such a protocol guides the agents through the decision-making, helping them to engage in an informed and justified process by, for example, suggesting that they consider their needs and objectives, and available resources.

When we discuss protocols it is useful to consider three principal categories of negotiation: (1) decision and choice; (2) language; and (3) process (Muller, 1996). Each category addresses a different question:

1. What to communicate?
2. How to formulate the message?
3. When to present the message?

The categories, their relationships to the negotiation and their main constructs are illustrated in Fig. 1.

Decision and choice involve all activities that a negotiation participant undertakes individually and without involvement of her counterparts. These activities include the person's consideration of the relevant attributes and preferences, formulation of reservation and aspiration levels, and the specification of feasible and acceptable alternatives. They may follow the prescriptions of decision analysis and they may be supported with decision aids. In this category we also have individual activities directly pertaining to the negotiation, for example, strategy selection and decision about making concessions and their size.

Process refers to the structure or model of the negotiation process, focusing on the joint actions and interactions of the negotiators, which may also include their individual actions.

Language refers to the terms used to describe information; its purpose is to formulate the communication content. In face-to-face negotiation language may be informal and the communiqué's meaning may not be clear, so the negotiators spend much time on clarifying the intended message. In e-negotiations, and especially in negotiations conducted by and with software agents, the language has to be well structured and unambiguous.

The negotiation language *primitives* are terms indicating the state and/or action; for example, propose, request, answer, and refuse. The *object structure* is the configuration of primitives used to describe a negotiation concept, such as, act, offer, rejection and request. *Ontology* (or taxonomy, if ontology is not available) is used to formulate meaningful statements using primitives and objects.

Ontology may describe the domain of the subject of the negotiation, for example, it may be a comprehensive description of air pollution together with the possible remedies. This description includes the entities that cause and reduce pollution, the pollutants, their properties such as intensity of pollution, usage and costs, and the relationships among the entities. In such a case, an ontology can be used as domain knowledge, helping the negotiators to understand and formulate the problem, and construct and analyze solutions, and also to formulate messages and understand the messages sent by others, who use the same ontology.

An ontology may also describe the negotiation as domain, i.e., a comprehensive knowledge base of negotiations. This allows a user of such an ontology to follow its relationship and select desirable properties and attributes of negotiation concepts in order to achieve an agreement.

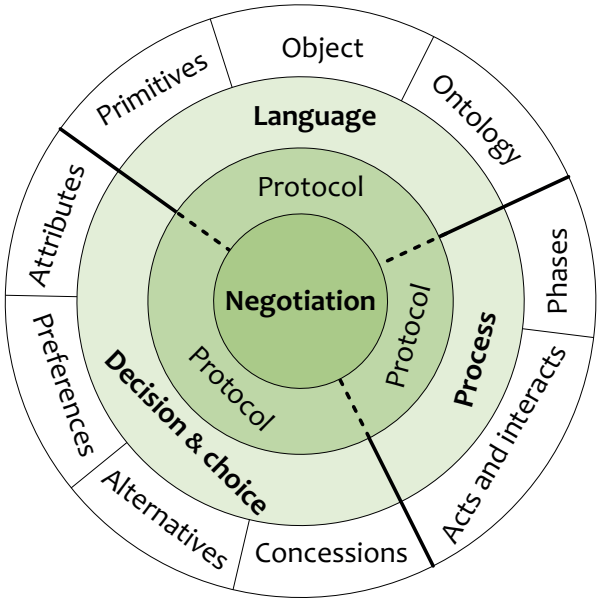

Figure 1. Three negotiation protocols and their key constructs. Adapted from Muller (1996), p. 213

The distinction between domain knowledge and process knowledge is important albeit in practice we must have a little (or more than a little) of both: we have to know the negotiation subject and we must know something about the process and its possible results. It is important, because a negotiation ontology can cover everything that pertains to the negotiation problems, processes, strategies, offers and so on. If we had such an ontology, then its positioning in Fig. 1 would be incorrect. This comprehensive knowledge of negotiation would include every possible negotiation protocol. Typically, however, there is a clear distinction between problem knowledge, process knowledge and language used for communication.

The construction of negotiation and other ontologies has been undertaken in the multi-agent system (MAS) community. Ontologies can provide the general framework for software agents' participation in negotiations. The agents can use it to view and compare protocols that are implemented in this ontology and decide on the one that fits best the particular type of negotiation they need to conduct (Tamma et al., 2005).

The construction of a negotiation ontology is a large and difficult enterprise. Several ontologies have been proposed, but they are very narrow in scope and applicable for only research and testing of software agents' behavior (Dong, Hussain and Chang, 2008).

The partial taxonomy, which I discuss in this chapter, indicates the scope of such an endeavor and its difficulty. A possible approach is to do it in stages and in a piece-meal fashion. The downside of such an approach includes the necessary overlapping of the results, and the introduction of contradictions and redundancies. But this would give us ontologies and taxonomies narrowly focused on one or a few negotiation types, as well as protocols serving different purposes. These results could immediately be tested and compared, leading to more comprehensive taxonomies, ontologies and protocols.

Out of necessity, researchers and designers take a narrow and focused approach to the construction of taxonomies, ontologies and protocols. This perspective is reflected in Fig. 1; the ontology scope is limited to the content of communication, it helps the users to understand and agree on meaning of terms and messages. There are three separate protocols indicated, each responsible for the organization of the activities associated with the respective category.

Ontologies and protocols are required for flexible and capable negotiation software agents. They are also very useful in the design of ENSs and agents collaborating with and helping human negotiators. There is large specialized literature on multi-agent systems and their interactions, therefore in the next section I only briefly mention some of the developments in this area and present a very simple communication protocol.

1.1.3. Ontologies and language protocols

Software agents that engage in fully autonomous negotiation, cooperation and other social processes require the capability to undertake purposeful actions, select strategies that guide these actions, and effectively communicate with others (Ermolayev and Keberla, 2000). The agents that are able to access and use a common ontology can engage in negotiation without being specifically programmed to do this, exchange information about the negotiation mechanism, formulate a shared view and agree on the meaning of concepts (see Tamma et al., 2005, p. 225).

An example of an architecture that includes negotiating agents who have decision-making skills and are able to search for, choose and use the most suitable communication protocol, is presented in Fig. 2. This high-level architecture is proposed by the Foundation for Intelligent Physical Agents (FIPA) as the ontology service reference model; its purpose is for the agents to operate in open environments. The reference model is adapted here for the negotiation context.

The three negotiating agents, shown in Fig. 2, request services from a dedicated ontology agent. The services include location of ontologies that are relevant for the agents' tasks, translation of different ontologies' expressions and terms, analysis of differences between terms, translation of different content languages and communication protocols (e.g., OKBC, OQL, HTTP) into the one that is preferred by the negotiating agents.

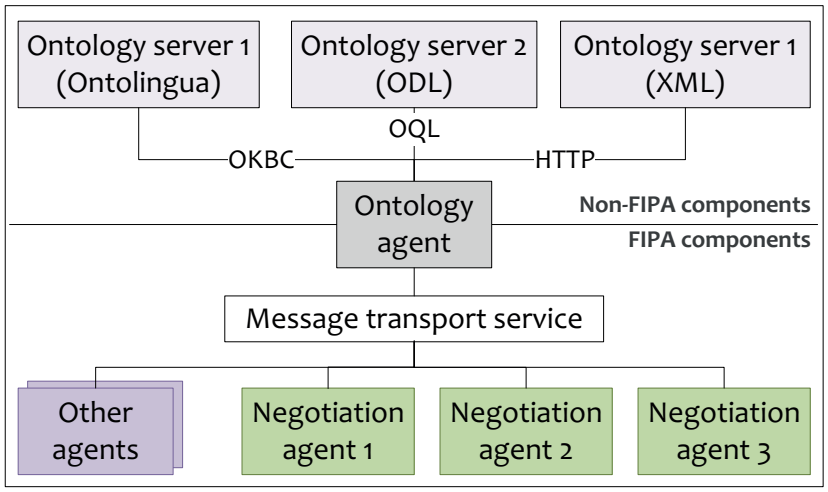

Figure 2. Ontology services for software agents' negotiations (adapted from FIPA, 2000, p. 7)

The agents access the ontologies (via the Ontology agent) in order to engage in the ontology-based communication, which is controlled by the agents' strategies and tactics and the language protocols.¶

A very simple language protocol is illustrated in Fig. 3; it is an example of an interaction protocol proposed by FIPA (FIPA, 2002).

The protocol determines that the transaction initiator prepares a proposal during activity A. This activity is completed when the initiator sends a message with a proposition of performing a certain action. The participant analyzes the proposal and may also perform other actions (B), which end when the participant responds with either an acceptance of the proposal or its rejection. Following the receipt of the response, the initiator undertakes either actions C_1 or C_2.

Similar protocols have been designed for various auction and negotiation mechanisms. They are often significantly more complex than the protocol shown in Fig. 3; have many more objects, interactions, and participants, but the idea is the same.

¶These types of protocols should be called communication rather than language protocols. Because the former term is widely used in network and computer-to-computer communication, terms like interaction protocol and language protocol have been used in communication among software agents.

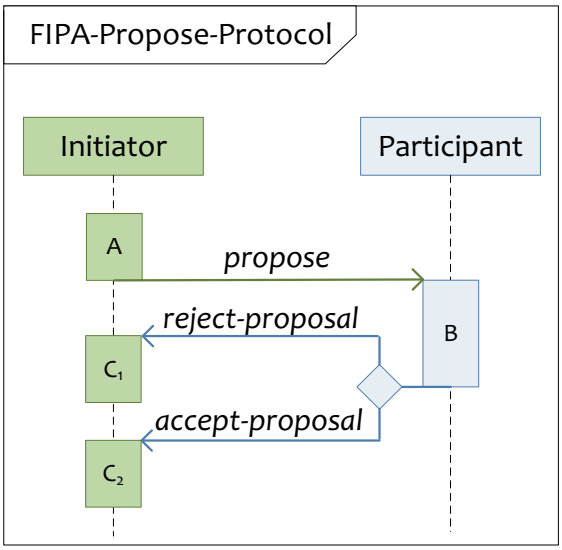

Figure 3. Language protocol example (from FIPA, 2002)

1.2. Negotiation protocols and their characteristics

Negotiation protocols have two meanings. In diplomacy, a protocol prescribes the permissible rules of interactions—the etiquette. The second meaning refers to the document describing a successful discussion, an agreement, various clauses that govern the agreement implementation, and permissible actions which the parties may take in order to enforce the agreement or when its implementation fails. The use of a software in negotiation requires that this software undertakes known and defined activities.

1.2.1. Definition

The negotiation phase model provides an overall framework within which activities pertain to the formulation of constructs and that the constructs themselves can be positioned. The process of construct formulation that takes place during e-negotiations is governed by the negotiation protocol. Every ENS component that receives information has to "know" how to process it and what to do with the output; it must follow a pre-defined set of communication and processing rules.

Computer scientists working on the design of automated negotiations proposed that the software agents' behavior be controlled with a negotiation protocol, see Cranor and Resnick (2000), Jennings et al. (2001), or Benyoucef and Keller (2000). The scope and degree of control of the interactions between users and the ENS need not be as complete and detailed as in automated negotiations.

Some level of the organization of the processing and communication tasks is, however, necessary. Thus, every ENS needs to follow rules and procedures embedded in its protocol.

E-negotiation protocol is a process model guiding information processing and communication tasks of the entities, participating in the process.

Protocols impose—explicitly or implicitly—restrictions on their activities through the specification of permissible inputs. Their two key functions, guiding processing and communication tasks, have causal character. This means that the allowable tasks and activities in one state are dependent on: (1) the tasks undertaken in the earlier states, (2) the situation in the current state, and (3) the expected future states. This causality can be represented with rules or a model that is equivalent to the rule-based model.

The restriction on the permissible inputs may be explicitly implemented in the protocol, or it may be associated with the state in which this input is required. In the latter case, the protocol rules activate the negotiation state in which the restrictions are formulated. This implicit specification of restrictions is advantageous, because it increases the flexibility of the protocol application. The same protocol can be used for different e-negotiation processes. For example, the protocol need only invoke the offer construction state; the particular manner in which the offer is entered is then handled by the formula used in the offer construction state.

Negotiation protocols are not necessarily explicitly specified. In many early ENSs, and also in some recent systems, the protocols are implicit; their users must follow a particular implementation of the e-negotiation process. To be able to move from one activity or task, they have to provide information that the system requires. Neither the system nor its users can choose an activity but have to follow the pre-defined "hard-wired" sequence. For each of these systems, however, it is possible to re-construct and formally represent the protocol that the system and its users follow.

The simplest example of an implicit negotiation protocol is the one used in an e-mail system. Its users can communicate using text, and possibly attachments containing text, charts and other graphics. In this way the system restricts the permissible inputs. It places every message in a specific place, sequences messages, and notifies the user about new messages. The protocol is very simple and known to every email user.

Protocols may be implicit or explicitly specified. Implicit protocols do not allow their users to control an ENS execution. The protocol is implemented in a control program, which decides what component and when a component is activated. Explicit protocols display rules, activating components, and let users activate these rules.

Systems with active components allowing their users to control the components' sequencing and execution need explicit protocols. For example, the

user may be able to submit one, two or more offers at the same time; she may submit a complete package or only values of selected issues; or she may make a conditional offer. Similarly, the system may suggest one or more packages; propose that no offer be made; or suggest loosening soft constraints that restrict the values of one or more issues. In these situations the selection of a particular solution opens a new path of contiguous activities; every possible path selected by the user and/or system has to lead to the desired negotiation outcomes.

1.2.2. Protocol design

At present, knowledge about protocol design comes from two very different sources. The first source is artificial intelligence and software engineering. While these two areas have little in common with negotiations among people, there has been a significant amount of theoretical and applied research on the cooperation and negotiation among software agents, see Jennings (1995), Kraus (2001), or Zlotkin (1996). Practical applications focus on the division and allocation of tasks and resources among computers and other systems (e.g., industrial printers, databases and robots).

The second source of knowledge is social science, in particular, all its areas concerned with behavioral research on negotiations and negotiators. Because there have been no studies of actual e-negotiations and there are only preliminary results from experimental research based on negotiation teaching, the protocol designers have to adapt results from face-to-face negotiations to e-negotiations. This is a tentative approach, because e-negotiation participants may exhibit behaviors, use strategies and tactics, and undertake activities that are quite different from those employed in face-to-face negotiations. It also poses difficulties for the designers, because they need to construct formal representation of often imprecise and vague prescriptions and ill-defined negotiation constructs.

The Montreal taxonomy (Ströbel and Weinhardt, 2003) and its extension given in Section 3[‖], provides ENS designers with the terminology, a description of the negotiation constructs, and their roles in the process and the relationships between them. This taxonomy allows us to describe the process and justify its activities. This, in turn, makes it possible for the e-negotiation participants to: (1) select the specific e-negotiation they wish to conduct; (2) know what the system is doing and why; and (3) understand their tasks and how these tasks may contribute to the negotiation outcomes.

In face-to-face negotiations the decision regarding the protocol, most often only partial, is solely in the hands of the negotiators and/or those on whose behalf they negotiate. The activities the negotiators undertake depend on their own and the counterparts' characteristics and orientation, the strategies and tactics they choose and the negotiation phase (see Fig. 4).

[‖]This is the reference to the Section 3 in Gregory Kersten's entire book (eds.)

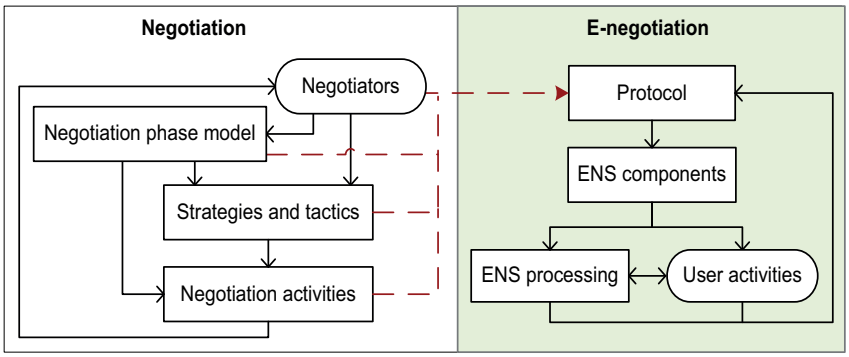

Figure 4. Face-to-face and electronic negotiations

The relationship between negotiation and e-negotiation is illustrated in Fig. 4. The information that causes an e-negotiation activity to be undertaken is embedded in the protocol. It is the information which negotiators use purposefully but it also may be the information that describes their characteristics and perceptions, and which influences their activities.

The negotiation protocol selects the system's components which process data and/or request input from the user (Fig. 4). Information about the completion (successful or unsuccessful) of an activity is passed to the protocol, which selects the next activity. In this way the protocol controls the ENS and guides the users' activities.

The relationship between strategies and tactics in negotiations and the protocol in e-negotiation is associated with the consideration of the negotiation protocol as the top-level component that organizes activities of all ENS components, oversees activities of ENS users, and synchronizes activities of both users and system components. At this level there is no indication of ownership; a part of the protocol may be owned by and/or accessible only to one user and not others so that she can protect her information and undertake actions which others are not able to see.

1.2.3. Protocol types and properties

Several types of protocols are possible; a list of different protocols and the short explanations are given in Table 1.

In the consideration of closed and open protocols we need to introduce two perspectives: the *user perspective* and the *system perspective*. Systems that support only completely structured exchange of information and disallow exchange of free-text messages are closed from both these perspectives. Examples of these systems are auction systems, in which the parties may submit only the

issue values. Early ENSs such as NEGO (Kersten, 1985), RAINS (Hordijk, 1991) and Web-HIPRE (Mustojaki and Hämäläinen, 2000) are examples of systems in which implicit closed protocols were implemented. Closed protocols are also used in automated negotiations; with the negotiation software agents following the predefined set of rules.

Table 1. Types of e-negotiation protocols

Type	*Description*
Closed	All rules are defined a priori; no rule can be added or modified
Open	Rules may be constructed and added during the negotiation
Private	Guides the user's activities and defines her valid actions
Public	Defines the rules of interactions between the negotiators
Comprehensive	Can be used for different types of negotiations
Specialized	Applicable to one or a few negotiation types

A system that allows for exchange of free-text messages may follow a closed protocol. Its users, however, may introduce and agree upon new rules, which they follow irrespectively of the rules that the system follows. For example, users of the Inspire system may ignore the system's components used for offer exchange and conduct the negotiation solely via the exchange of messages (Kersten and Noronha, 1999). The system does not recognize these messages as offers and it cannot use its analytical and graphical components. If the users achieve an agreement, Inspire does not recognize it and thus cannot assess its efficiency. In effect, while Inspire follows a closed protocol, users may follow an open one.

From the user perspective most protocols of the recently developed ENSs are open. With the exception of systems for on-line auction and bidding systems this trend will continue, because of the need to provide negotiators with systems that allow them to engage in unrestricted discussions.

From the system's and thus, the system designer's perspective an ENS with open negotiation protocol requires a facility for the user or facilitator to construct and add new rules. Open protocols add complexity in the system construction and use, but they may be required to account for the negotiators' learning and encountering problems that cannot be addressed with any existing rule.

It is useful to distinguish between private and public rules. Private rules are used to, for example, educate the negotiator, and help her select a strategy, evaluate counter-offers, make concessions and formulate arguments.

Public rules are used to set up the agenda, the kind of deals that the participants can make, message structure, and allowable sequence of offers and

counter-offers. They may include the requirement that the negotiation is conducted in good faith or that an issue that both parties agreed upon cannot be renegotiated.

Another distinction of protocols is with respect to their comprehensiveness; comprehensive protocols are those which can be used for several different types of e-negotiations. They allow using the ENS, in which they are implemented, for different negotiation processes and problems. There are few ENSs that implement comprehensive protocols and most of them deal with auctions (Ströbel, 2003; Benyoucef et al., 2001). INSS was an early attempt to construct a system for several negotiation types. The Invite platform is an example of software capable of supporting many negotiation types.

A specialized protocol is the one that describes one or a few negotiation types and problems. Examples include SmartSettle, Inspire and WebNS.

E-negotiation protocols can be characterized by several desirable properties presented in Table 2.

Table 2. Properties of e-negotiation protocols

Property	*Description*
Input consistency	All available information is considered for processing
Transparency	Users can observe and understand ENS behavior and actions
Explicability	Reasons for action selection are justifiable
Tractability	The purpose of every potential activity can be justified
Satisfiability	The protocol is not inconsistent; it is possible to reach a concluding state of the negotiation (e.g., an agreement or breakdown)
Completeness	Interactions between the users and the ENS are sufficient to achieve the goal of the negotiation.

Input consistency means that no information, available to the system, can be ignored even if this means the need to resolve possible input inconsistencies. For example, if BATNA and the reservation levels are available, then they are both considered in assessing a counter-offer.

Protocol *transparency* is required for users to know what activities the system is undertaking. It does not necessarily mean that the user understands the role of every element of the protocol and every component of the system. Elements may be grouped together jointly and applicable to one negotiation activity. Transparency also means that the user understands the sequence of the system's activities and can position each activity in the negotiation phase model. For example, a set of rules used to assess the counter-part as a hard negotiator is linked (through the opponent's assessment) with a set of rules used to make a concession.

The *explicability* property ensures that the activity selected by the protocol is traceable and the system can justify it.

The *tractability* property is more general than explicability; it refers to the protocol's capability to justify all activities that can potentially be undertaken.

The *satisfiability* property ensures that the negotiators, who begin the negotiations using a given protocol, are able to reach a state that—according to the protocol designers—concludes this negotiation. In other words, there is at least one such sequence of states that leads from the negotiation opening state (beginning) to a final, negotiation termination state, either an agreement or a breakdown. It is a weaker property than completeness, which assures that every path that begins with the initial state leads to a final state.

The *completeness* property means that the activities undertaken by the user and the system under the guidance of the protocol lead to the negotiation goal, which is the achievement of the agreement or the realization that an agreement cannot be reached and the negotiation has to be terminated. Although this second goal may be considered a failure, it is important that the protocol allow for the process termination.

The completeness property assures that there are no "gaps" in the protocol and that for every negotiation situation there is an activity that can be undertaken so that the process continues. For every state, in which the negotiators and the systems find themselves, there is always at least one sequence of activities leading to the negotiation conclusion. This property also assures that there is no activity that can be undertaken by the system that cannot be invoked by the protocol.

1.3. Phases, states and activities

E-negotiations and negotiations alike can be seen as a sequence of activities. The purpose of an activity is to formulate or reformulate the negotiation construct to which it pertains.

Some of the activities are necessary; they must be undertaken by the negotiators or the system for the negotiation to take place. For example, the negotiator has to consider (learn about) the problem, formulate and propose offers, assess counter-offers, and accept or reject a compromise proposal. Other activities may, but do not have to be undertaken. For example, the negotiator may send a message, attach an explanation to the offer or define aspiration levels.

Every activity takes place in a given *negotiation state* that belongs to one negotiation phase. In e-negotiations the activities are undertaken by the negotiators and by the systems they are using. Some activities may be combined, for example, the system may request that the user make an offer and write its justification, or the negotiator may, at the same time, view the counter-offer and read the negotiation problem. This introduces the possibility that the sys-

tem and the user are at the same time in different states. Furthermore, in one ENS, some activities may be combined, while in another system they may be considered separately.

An example, illustrating the relationship between the negotiation phases, sequences, activities and outcomes is shown in Fig. 5.

A phase is decomposed into sequences of similar or complementary activities. In every sequence one or more activities are undertaken. In general, the activity's result is an output specifying one or more values of an outcome or which enables the undertaking of another activity. There may also be activities that coordinate the process or are required for user-system interaction (e.g., the negotiator's confirmation of the offer submission and logging out from the negotiation).

One or more activities may be used to obtain the same outcome. In Fig. 5, Activities 1, 2 and 4 are used to formulate Outcome 1.

At any point of the negotiation the user and the system are in the state of performing an activity from a particular sequence. Completion of all activities, associated with this state, moves the negotiation to another state. The completion of Activities 1 and 2 moves the negotiation to Sequence 2 and the completion of Activity 2 in Sequence 2 moves the negotiation to Sequence 3. As can be seen, the same activities may appear in different sequences, for example, in two or more states the negotiator may write a message to her counterpart or read the negotiation problem.

Completion of some activities creates a situation when more than one move is possible. Upon completion of Activity 3, the move to either Sequence 4 or Sequence 2 of the preceding phase is possible (Fig. 5). This allows the negotiator to cycle through a series of the same states; for example, the negotiator adds a negotiation issue, then formulates options for this issue, adds another issue, and so on.

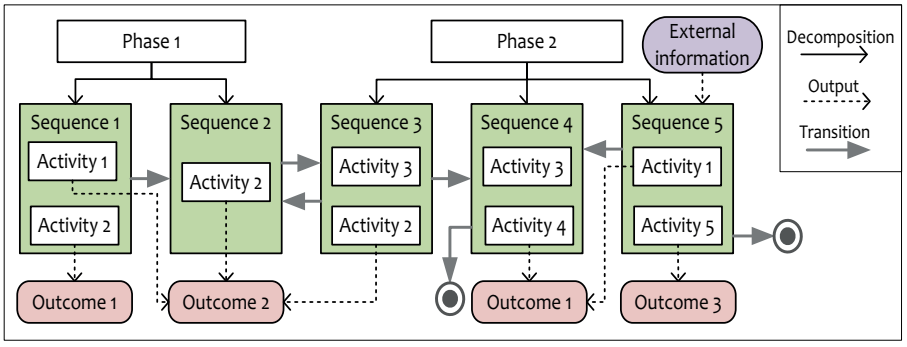

Figure 5. Phases, activities and outcomes

The moves between phases and states (Fig. 5) represent the process from the perspective of one negotiator. A similar representation may be constructed for other negotiators. The exchange of information between the negotiators invokes activities that the negotiator may undertake; external information activates Sequence 5, in which Activities 1 and 5 are undertaken. For example, an offer submitted by a counterpart invokes a state in which the negotiator evaluates this offer (Sequence 5). Based on the counter-offer evaluation, the negotiator moves to the state in which she constructs and submits an offer (Sequence 4).

From the above description of the activities undertaken in various negotiation sequences it follows that sequences may be visited more than once. It also follows that the same activity may be undertaken in several sequences and the completion of an activity leads the negotiator and/or the system to the next sequence. An exception is the set of activities undertaken by the negotiator's counterparts.

The activities undertaken by the counterpart (e.g., activity "submit offer") are associated with the counterpart's state. However, the information they produce activates the negotiator's state (e.g., in Fig. 5, Sequence 5 is activated by the information provided by the counterpart).

I mentioned above that some negotiation activities have to be undertaken and others may, but do not have to be undertaken. This distinction, together with the distinction between the negotiator's and the counterparts' activities, allow for categorization of negotiation states into the following three types:

1. *Mandatory sequences* comprise activities which the user or the system has to undertake;
2. *Optional sequences* comprise activities that may but do not have to be undertaken; and
3. *Intervening sequences* are activated by information that is external to and not controlled by the user or the system.

The distinction between the three sequence-types is context-dependent. In the example presented in Fig. 5, Sequence 2 is mandatory for Sequence 1, that is, the completion of Activity 1 moves the negotiation to Sequence 2. However, when the negotiation is in Sequence 3, then Sequence 2 becomes optional; the negotiator (or the system) may either move to Sequence 2 or to Sequence 4. The negotiator may cycle between Sequence 2 and 3, but to move to a new state at some point she has to select Sequence 4. Thus, Sequence 4 is mandatory for Sequence 3.

References

Benyoucef, M., Alj, H., Vezeau, M. and Keller, E. K. (2001) Combined Negotiations in E-Commerce: Concepts and Architecture. *Electronic Commerce Research Journal*, **1**, 3, 277-299.

Benyoucef, M. and Keller, R. K. (2000) An Evaluation of Formalisms for

Negotiations in E-Commerce. In: *Workshop on Distributed Communities on the Web*, Quebec City, **LNCS 1830**, Springer Verlag, 45-54.

CRANOR, L. AND RESNICK, P. (2000) Protocols for Automated Negotiations with Buyer Anonymity and Seller Reputations. *Netnomics,* **2**, 1, 1-24.

DE DREU, C. K. W., WEINGART, L. R. AND KWON, S. (2000) Influence of Social Motives on Integrative Negotiation. A Meta-analytic Review and Test of Two Theories. *Journal of Personality and Social Psychology,* **78**, 5, 889-905.

DONG, H., HUSSAIN, F. K. AND CHANG, E. (2008) State of the Art in Negotiation Ontologies for Enhancing Business Intelligence. In: *4th International Conference on Next Generation Web Services Practices*, Seul, IEEE, 107-112.

ERMOLAYEV, V. AND KEBERLE, N. (2006) A Generic Ontology of Rational Negotiation. In: D. Karagiannis and H. C. Mayr, eds. *Information Systems Technology and its Applications*, **84**, *Lecture Notes in Informatics*. Bonn: Gesellschaft für Informatik, 51-66.

FIPA (2000) FIPA Ontology Service Specification, FIPA, Geneva. [Online]. Available: http://www.fipa.org/specs/fipa00086/ XC00086D.pdf

FIPA (2002) FIPA Propose Interaction Protocol Specification. FIPA, Geneva. [Online]. Available: http://www.fipa.org/specs/fipa00036/SC00036H.html

HORDIJK, L. (1991) Use of the RAINS Model in Acid Rain Negotiation in Europe. *Environmental Science Technology*, **25**, 4, 596-603.

JENNINGS, N. R., FARATIN, P., LOMUSCIO, A. R., PARSONS, S., WOOLDRIDGE, M. J. AND SIERRA, C. (2001) Automated Negotiations: Prospects, Methods and Challenges. *Group Decision and Negotiation*, **10**, 2, 199-215.

JENNINGS, N. (1995) Controlling Cooperative Problem Solving in Industrial Multi-agent Systems Using Joint Intentions. *Artificial Intelligence*, **75**, 2, 1-46.

KERSTEN, G. E. (1985) NEGO - Group Decision Support System. *Information and Management,* **8**, 5, 237-246.

KERSTEN, G. E. AND NORONHA, S. (1999) Negotiations via the World Wide Web: A Cross-cultural Study of Decision Making. *Group Decision and Negotiations*, **8**, 251-279.

KRAUS, S. (2001) *Strategic Negotiation in Multiagent Environments.* MIT Press, Cambridge.

MULLER, H. J. (1996) Negotiation Principles. In: G. M. P. O'Hare and N. Jennings, eds., *Foundations of Distributed Intelligence.* Wiley, New York, 211-230.

MUSTAJOKI, J. AND HÄMÄLÄINEN, R. P. (2000) Web-HIPRE: Global Decision Support by Value Tree and AHP Analysis. *Information and Management*, **38**, 3, 208-220.

STRÖBEL, M. (2001) Design of Roles and Protocols for Electronic Negotiations. *Electronic Commerce Research Journal,* **1**, 3, 335-353.

STRÖBEL, M. (2003) *Engineering Electronic Negotiations.* Kluwer, New York.

Ströbel, M. and Weinhardt, C. (2003) The Montreal Taxonomy for Electronic Negotiations. *Group Decision and Negotiation*, **12**(2), 143-164.

Tamma, V., Phelps, S., Dickinson, I. and Wooldridge, M. (2005) Ontologies for Supporting Negotiation in E-commerce. *Engineering Applications of Artificial Intelligence*, **18**, 2, 223-236.

Zlotkin, G. (1996) Mechanisms for Automated Negotiation in State Oriented Domains. *Journal of Artificial Intelligence Research,* **5**, 163–238.