Towards Automating Negotiation for M-Services

Shamimabi Paurobally, Phillip J. Turner and Nicholas R. Jennings
University of Southampton, Dept. of Electronics and Computer Science, Southampton SO17 1BJ, UK.
{sp, pjt, nrj}@ecs.soton.ac.uk

ABSTRACT

Mobile electronic commerce (m-commerce) is an emerging manifestation of internet electronic commerce that bridges the domains of Internet, mobile computing and wireless telecommunications in order to provide an array of sophisticated services (m-services) to mobile users. In such an open market place, the flexible exchange of m-services will necessarily be through some form of negotiation. To this end, this paper investigates the issues associated with such negotiation in facilitating agent mediated mobile commerce (many of which also apply to the wider electronic commerce domain). Specifically, we discuss the protocol and strategies for evaluating and generating sets of issues when negotiating for a service in a future generation mobile telecommunication environment.

1. INTRODUCTION

Future generation mobile telecommunication systems (3G and beyond) are increasingly being viewed as open marketplaces in which the various stakeholders produce and consume services [11]. Examples of such m-services are mobile shopping (booking a flight before departure, then reserving a car on arrival), location-sensitive information (map services, local hotels and weather information), telemetry (traffic updates and logistics tracking) and mobile banking (service billing and stock market transactions). As these examples imply, there are a number of aspects to the trading of m-services, including customisation, personalisation, location-sensitivity and context-awareness [11]. In order to offer services with such properties and, at the same time, to be effective at the speeds and capacities available in wireless systems, the processes of service discovery, provision and execution need to be automated. In particular, flexible interaction mechanisms such as automated negotiation enable trading in dynamic and unpredictable environments, whilst allowing the involved participants to deal with different and conflicting preferences and goals. However, we believe that future generation telecommunication systems will have a number of limitations and constraints that will affect agent negotiations and this is where the main focus of the work lies.

Automated negotiation may be defined as a form of decision making where two or more agents jointly search a space of possible solutions with the goal of reaching a consensus. There are many different forms of negotiation (including auctions and bi-lateral encounters) that are tackled using many different techniques (including game theory, heuristics and argumentation) (see [3] for an overview). Here, however, we focus on bi-lateral encounters (since we found these to be especially prevalent in this domain [14] and we consider the sequential alternating offers model (Rubenstein) (again for reasons of commonality of use).

Against this background, this paper specifically investigates the mechanisms for automating negotiation for m-services in a future generation mobile telecommunications environment. In such an environment, resources may be scarce or costly, and access to them may be sporadic. Therefore, when evaluating and generating exchanged sets of negotiation issues, there are various factors that should be considered. For example, the quality of the underlying communication network, an agent's resources and its interaction experiences. In this paper, we discuss various open issues that have come to light during our efforts to automate m-service negotiation. These include finding appropriate evaluation and generation functions that reflect a user's preferences, the dynamics of these functions and reconciling existing work on strategies with richer negotiation protocols.

Some approaches to automated negotiation assume perfect rationality of agents, where the strategies and the best actions are computed instantaneously [10]. Although these lead to important theoretical contributions to negotiation strategies, their properties, optimality and equilibrium in constrained environments, the underlying assumptions are often inappropriate for practical contexts. Thus, this work seek to develop models that can be used in practice (the downside of which is that results tend to concentrate on typical performance with few guarantees and outcomes need to be determined empirically).

In more detail, this paper contributes towards a pragmatic approach for automating m-services negotiations and discusses some of the issues we have encountered. We concentrate on ways to implement m-service negotiation and obtain empiral results. We do not assume that participants adopt a single dominant strategy because we are interested in how to design the various strategies that allow participants to bargain. A service-oriented model is adopted (as in [2]) for selling and buying services (as opposed to sharing resources). Another novel aspect is that the bilateral protocol used allows more choices in speech-acts than a more standard alternating offers model. Moreover, current research in AI negotiation and agent interaction tends to separate the design of intelligent strategies and the specification of agent communication languages and protocols. They focus on either one or the other. We seek to bridge this gap with realistic strategies and realistic protocols.

The remainder of the paper is structured as follows. Section 2 discusses m-commerce transactions and presents an m-commerce scenario that was motivated by this work. Sec-

AAMAS-03 Workshop on Agent-Mediated Electronic Commerce V (AMEC-V) 2003 Melbourne, Australia.

tion 3 specifies the bilateral negotiation protocol we use. Section 4 and 5 discuss the issues arising when designing strategies for bilateral negotiation and proposes some solutions. Section 6 gives our proposed methodology for combining the protocol with the strategies. The paper concludes with section 7 which includes further open issues.

2. M-COMMERCE TRADING

M-commerce is concerned with the set of applications and services accessible from Internet-enabled mobile devices. It has requirements over and above those of more traditional e-commerce, including services that are accessible over wireless networks and that are adaptable according to the characteristics of the mobile devices for which they are configured and run on. In particular, limited screen sizes, low data rates associated with mobile Internet devices, rapid deployment of accurate location-tracking technology, as well as the time critical nature of many of the tasks mobile users engage in, are likely to contribute to the increasing demand for mobile location-sensitive services [11]. Thus we believe that as technology and infrastructre progress, location-sensitive and context-aware services will become increasingly prevalent.

Given the limitations of mobile devices, m-commerce requires well-targeted and concise content presented to the user, relevant to their preferences, locations and activities. To achieve such ends, negotiation can be used in the trading of both telecommunication and high-level services, where the network may deteriorate with low bandwidth, bounded coverage, latency, fluctuating error rates and spurious connections. The scenario in the next sub-section illustrates an electronic transaction involving interactive forms of negotiation across mobile and static devices.

2.1 An M-Service Scenario

Figure 1 shows the scenario's participants – customers, vendors, suppliers, banks and facilitators. The labels i_n on the arrows represent information flows between entities as they negotiate with each other for services. For example, Mr. Smith is on a train and his agent, acting as a customer on a mobile phone, browses and negotiates the offerings of one or more vendors (the information flow *i1*), for purchasing flight tickets.

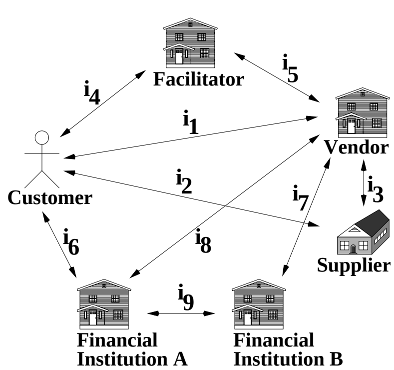

Figure 1: M-Commerce scenario involving negotiation

1. Mr. Smith's agent, Agent Smith, (from his mobile phone on a train) negotiates for the brokerage services of a facilitator (interaction *i4*) by deliberating on the facilitator's cost, effort to be spent on finding vendors and the deadline for finding them. Agent Smith provides the transaction requirements to the Facilitator agent, such as departure point, time, destination and maximum price.

2. Facilitator negotiates for the services of various vendor agents for information about flights for the requested route (*i5*). It selects Vendor1 as offering the most suitable package and further negotiates with it. Facilitator submits Mr. Smith's payment credentials to Vendor1.

3. Agent Smith negotiates for a loan with a banker at financial institution A (*i6*) regarding the amount, interest rates and time of repayment. The banker is in a seminar and his agent is active on his laptop via a wireless link. Since Mr Smith has a good bank record and the loan is not high, the banker's agent does not alert its owner.

4. Upon Vendor1's request, Financial institution B's agent contacts Financial institution A's agent and determines the ways to have funds transferred (*i9*).

5. Vendor1 negotiates with the Supplier agent on how to deliver the tickets (*i3*). The Supplier agent negotiates with Agent Smith over the most appropriate way and time to deliver the tickets (*i2*).

6. Vendor1 compensates Facilitator for the completed transaction (*i5*).

As can be seen, the interaction between the participants in this scenario incorporates negotiation at various stages. Some of the participants are using mobile devices and are located on trains, walking or in a seminar. To support such a scenario and the dynamic trading of m-services, m-commerce architectures must consider issues such as scalability, microtransactions, reliable functionality (transient connections, failures, reduced network performance and degradation), fraud prevention and detection. However, in this paper, we focus specifically on the constraints on the negotiation for a service (rather than on the mobile constraints on services such as maps and service discovery).

2.2 M-Service Negotiation

Like all other environments, negotiation in m-commerce environments needs to consider the constraints and requirements imposed by the domain. For example, agents on mobile devices would normally be involved in short negotiations whereas those on fixed hosts can take part in continuous and computationally expensive negotiations. Mobile phones requires well-targeted and concise content presented to the user and therefore the m-service feature negotiation would not include presenting animations, banners and long lists of results. Moreover, the features of mobile telecommunications determine the suitable negotiation mechanism and are often inter-related; the quality of of the network, QoN, may itself be partially defined in terms of the quality of service measurements, QoS. The QoN in turn vary with changes in bandwidth, range, frequency of disconnections, costs of connection, data integrity and security [1]. Negotiation mechanisms can be designed to adapt to such variations in bandwidth and QoN, for example:

- An agent's rates of concession or its decisions are influenced by the QoN. For example, if the QoN is low then an agent agrees to the first acceptable offer, while if the QoN is high, then an agent tries to bargain, search for the best deal and maximise its profit.

- Bandwidth limitations and fluctuations restrict the number of users involved in a negotiation, the number of messages or the number of concurrent negotiation threads.

- If a mobile phone's or laptop's battery power is low, an agent may decide to concede and quickly find an agreement or send notifications that it will soon suspend or abandon the negotiation.

- If there is not enough computational memory and processing power, then an agent can choose not to adopt complex strategies. In cases of increased latency and loss of network performance, an agent may choose to timeout or increase its time to compute its strategies and plans while waiting for a message.

The above features may be considered in designing negotiation strategies for the agents to exhibit network-aware adaptive behaviours in order to achieve their goals or even acquire better deals than normally. In the following sections, we discuss how feasible such types of negotiation mechanisms are using existing strategies.

3. NEGOTIATION PROTOCOL

In general, protocols are used to coordinate the activities of a group of agents as they try to satisfy an agent's and group's goals. In this context, the requirements are for richer and more flexible protocols ([7], [8]) than those that are more routinely used in bilateral encounters [2], [13], [5]. Here specifically, there is a need to allow participants more scope in qualifying exchanged messages with a communicative speech-act. To this end, figure 2 specifies a bilateral protocol [8] between two agents looking for an agreement over a negotiation subject. The protocol allows requests, proposals, offers and agreements and may form the basis for further customisation to allow richer interactions. We propose using this protocol in our framework for m-service negotiation because it allows a number of speech-acts other than offer and counter-offer, yet is simple to understand and implement.

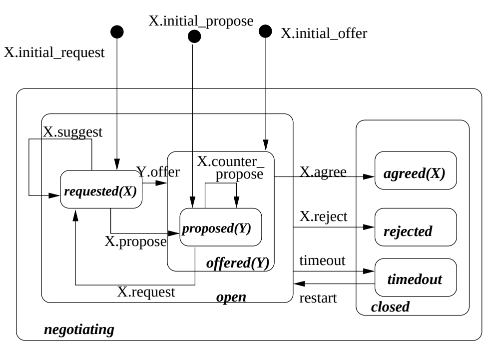

Figure 2: A Bilateral Protocol

Figure 2 gives the statechart of a bilateral protocol between agents X and Y. Entry into negotiation is through either agent sending/receiving an *initial_request*, *initial_offer* or *initial_propose* message which leads to an *open* state (more precisely to a *requested*, *offered* or *proposed* state respectively). For example, the process *X.initial_request* means that agent X sent an *initial_request* leading to a *requested(X)* state. The state *requested(X)* is read as agent X has triggered the state *requested*. Each state may be interpreted as conveying a level of commitment towards an agreement. For example an *agreed* state entails more commitment than a *requested* state. A *reject* action or *timeout* event can occur at any sub-state of an *open* state. From a *requested* state, both agents can continuously make *suggest* actions to remain in that state while modifying the subject of negotiation until one of them moves to a higher level of commitment via an *offer* or *propose*. The *proposed* state is a sub-state of *offered* and both of these states may allow an agreement to follow in the next action. This would result in an *agreed* state. (The difference between *offered* and *proposed* is that from the former state, an agent can only *agree* or *reject* whereas from the *proposed* state, an agent may *agree*, *reject* or return to a *requested* state through a *request*.) We also allow the two agents to restart a *timedout* negotiation through forking another bilateral negotiation with the negotiation subject being whether to restart the timed-out interaction. The protocol can be analysed for the properties it exhibits. Here an agreement is not guaranteed since it depends on the participants' strategies (as do most of the bilateral models that have previously been proposed).

4. NEGOTIATION STRATEGIES

Quoting from [15], "A key issue here is that, since we are interested in actually building agents that will be capable of negotiating on our behalf, it is not enough simply to have agents that get the best outcome in theory — they must be able to obtain the best outcome in practice." In this vein, there are various properties sought for during mechanism design such as guaranteed success, maximisation of social welfare, Pareto efficiency, individual rationality, stability, simplicity and fair distribution [10]. Moreover, it would be advantageous to replicate these properties in practical applications.

Towards this end, in this paper we discuss the strategies that can be used for evaluating and generating values for sets of issues when exchanging a service. For the sake of conciseness, we refer to values for sets of issues as sets of issues. In this context, the agents in the group are not negotiating for sharing a resource [5] or performing a joint task such as delivering parcels [6]. Although we could say the trading of an m-service is a joint task of exchanging the service, the competition is not in accessing the service but in its profitable purchase and sale. Therefore, we adopt the intuitive service-oriented approach and notation in [9] and [2], which embodies the concept of electronic transactions.

We consider a negotiation between two agents a and b over a changeable set of issues J. An issue j, $(j \in J)$, can take values between $[min_j, max_j]$, which define the domain, D_j, of a quantitative issue and are effectively the reservation values of an agent. The domain of a qualitative issue is defined as an ordered set of possible values as in $D_j = \{q_1, ..., q_n\}$. Let the term $x^t_{b \to a}$ denote the negotiation subject consisting of values associated to issues, sent from agent b to a at time t. The value of an issue does not depend on the other issues.

4.1 Meta-Strategies

There are various factors that influence an agent's negotiation strategy. In this context, a well-researched factor is time, namely the deadlines of the agent or the time taken to perform a negotiation. In addition to environmental factors such as time, a strategy may also focus upon behavioural factors such as an opponent's behaviour. Being in an m-commerce environment, we investigate the effect of the following environmental factors, leading to its namesake strategy:

- Time aware strategy $\mathcal{T}$. This strategy takes into account the current time, the agents' deadlines, the time they have left, the time to exchange and compute messages.

- Network-Aware strategy $\mathcal{Q}$. The quality of the network, QoN, is used in this strategy where QoN is measured in terms of bandwidth, latency, messages lost and delayed.

- Experience strategy $\mathcal{E}$. The experience of an agent influences its decision. For example, the following may influence an agent behaviour – knowing how long it took previously to negotiate with a similar opponent, knowing if agreements are usual with the latter and knowing how much profit it can expect. Moreover, the experience of an agent is affected by past and present negotiations or the group's experience in general.

- Priority strategy $\mathcal{P}$. An agent may be involved in a number of concurrent negotiation threads and may run the risk of being overloaded with messages. To increase its performance, an agent can assign priorities to the negotiation threads.

- Resource strategy $\mathcal{R}$. This strategy allows an agent to base its decision on the amount of resources it possesses, for example computational power, goods in stock or money. (In essence, all the above factors could be categorised as resources.)

- Hybrid strategy. Such a strategy consolidates different factors, strategies or tactics.

The strategies $\mathcal{T}$ and $\mathcal{Q}$, are useful in m-commerce since an agent has time constraints and to adapt to a varying network. The strategy $\mathcal{E}$ emphasises an agent's social environment, while the strategy $\mathcal{P}$ allows an agent to sift through parallel negotiation threads as is the case in real settings. The strategy $\mathcal{R}$ considers resources that may influence a transaction. Given the abundant amount of available strategies, we generalise our framework to support meta-strategies. A meta-strategy is defined as a strategy that allows swapping between strategies or combinations which have a weighed importance. Using meta-strategies allows us to generalise the specification and implementation of the evaluation/generation of issues without redundantly developing the same functionailty for each strategy.

4.2 Evaluation Functions

In a negotiation, participants exchange sets of issues (for example a sends $(x_{b\to a})^t$ to b at time t). Each agent has a scoring function $V_j^a : D_j \to [0,1]$ that gives the score agent a assigns to a value of issue j in the range of its acceptable values [2], [9]. A weight ω_j^a is associated by agent a to issue j to indicate its relative importance, where the sum of weights of all issues is 1. An agent can change the importance attached to an issue by changing the weights associated to that issue. The evaluation of $(x_{b\to a})^t$ involves summing the valuation (score) of each issue in the negotiation subject, that is agent a rates the issues set x as: $V^a(x) = \sum_{1\le j\le n} \omega_j^a V_j^a(x_j)$.

4.3 The Scoring Function

Negotiation over multiple issues raises a number of questions. Current game theoretic results [5] and [13] may be reused if the issues are independent. The sets of issues may also be dynamic where issues can be added or deleted during the course of the negotiation. One way of dealing with this is to assign an issue with weight 0 by default, which is increased when an issue becomes part of a negotiation subject. However in open systems, where new issues are introduced on encountering new agents, there is a problem of modeling the ontology of the issues and a user's preferences.

In fact, the question arises of how agent a models the scoring function V_j^a for each issue j. According to a particular service, there are a number of issues possible such as price, quantity, delivery, brand, quality, colour and many others, let alone new issues from other participants. Each agent has to personally model the scoring function for *each* issue. Here the key issues are how the preferences of a user for an issue are translated to a scoring function? How are the family of functions and rate of growth determined and chosen between? For example, figure 3 could represent the scoring function V_a for a seller a and issue j against the issue price x_j. Then the maximum price may be assigned a valuation of 1, while zero price brings 0 valuation. This is a monotonically increasing function of the order a^x where a represents the rate of preference. The scoring function for a buyer would be inversely proportional, since a buyer would normally prefer low prices.

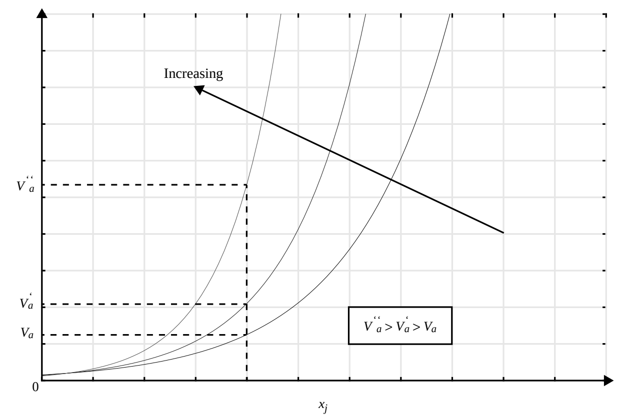

Figure 3: Scoring function for an issue

There are a number of open issues concerning the scoring function, where some related to modeling preferences:

- As mentioned above, for each issue, efforts may be required to design and optimise a scoring function, fixing or adapting the value of a, according to the preferences of a user.

- Are the scoring functions generic as in [2] or should they take into account the factors such as time, quality of network, etc.? That is, should a different evaluation function be designed for each strategy or resource? For example, an agent would evaluate a price p to be worth more when its deadline is close or when the network is slow.

- Qualitative issues could have quantified attributes, but this remains subjective to a user and may not be exhaustive.

Our work aims to address some of the above issues for monotonic functions. We also consider negotiations involving non-monotonic scoring functions for an issue and some of the problems we encounter are mentioned in the next section.

4.4 Non-Monotonic Evaluation Functions

An important consideration when evaluating or generating sets of issues is that scoring functions are often not monotonically increasing or decreasing according to a simple decay function. When the price of an item take values outside the reservation prices of the participants, what is its evaluation? The issue can be normalised where values greater than its maximum value are assigned a score of 1, regardless, and price 0 is assigned valuation of 0. There may however be cases of negative values for such issues. For example, a negative price implies a seller is paying a buyer to take the item.

Figure 4 shows various possible families of scoring functions. Some issues like quantity have a prefered value, so that its scoring function has a single maximum (line L4). Other issues may not have clearly defined or sensible maxima or minima, as for example, when discounting non-integer value for quantity (line L6). Moreover, we cannot rule out valuation functions with multiple maxima and minima where the valuation function is used to encode more than one desirable outcome (to bypass the need for chaining negotiation attempts on failure), e.g. L5. We note that various implementations of negotiation rely on the maximisation or minimisation of valuation functions (as appropriate) by defining a negative or positive dynamic to x_j instead of employing generalised hill-climbing over user defined valuation functions. Finally, each issue for each agent may have its own preference curve, possibly defined dynamically. Nevertheless an acceptable agreement would normally lie under the intersection of the preference curves of the negotating agents. When the utility functions are non monotonic, the following questions become pertinent:

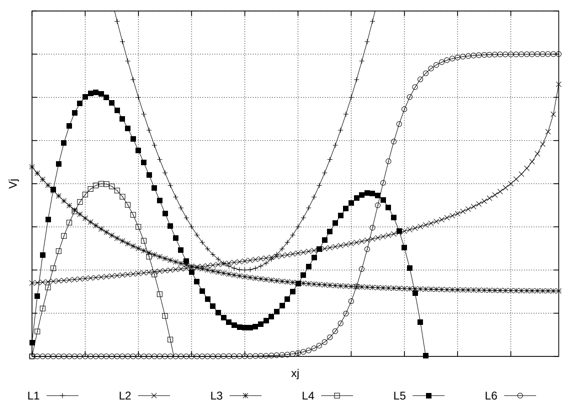

Figure 4: Dynamics of an issue

- Is it sensible to have minimum or maximum points, i.e. optimal preferences?
- What happens when an issue, such as quantity, has value 0, yet other issues are being negotiated upon?
- What do negative values mean? For price it may mean a reverse exchange of money, but what about quantity? Negative delivery time could mean the delivery has already been made and agents are finalising payment.
- The above bullet point raises the question of how to agree on the full details of the legality of an issue and its value. Is it possible for a badly encoded agent to allow a price tend to -∞ because the seller keeps offering more and more, or in another case +∞?

Evaluation of a set of issues is followed by the agent accepting, rejecting or responding with an alternative set of issues, that it generated using some strategy.

5. GENERATING SETS OF ISSUES

There are various strategies that can be re-used or adapted for the generation of a set of issues. For example, strategies considering the participants' deadlines, resources or opponents' behaviour are specified in [2] for calculating how much an agent concedes. They also define trade-offs strategies where an agent generates sets of issues of equivalent worth to itself, but of more benefit to its opponents.

In other cases, decision trees and assigning probabilities and utilities to the outcomes may be used. However, there may be the need to generate a whole tree or sub-tree towards possible final outcomes, which may be computationally expensive, where generating just the next response may be enough. In [10], monotonic concession strategies are given and their properties such as stability, deception-free or optimality analysed. In one such strategy, risk evaluation criteria is used for computing how much an agent is willing to risk by sticking to his last offer or for designing the agent with a smaller risk to make the next concession.

5.1 Reduction to a Single Offer

An alternating offers bargaining model is used in [4] and [6] for computationally limited negotiations. It is shown that the equilibrium strategies for the model results in a single shot take-it or leave-it strategy. The agents wait without exchanging offers until one of the agent's deadlines arrives and then the agent with the earlier deadline concedes and makes an offer that the other agent may accept. Figure 5 reflects this behaviour; it shows 30 concurrent negotiations between buyer and seller agents, with the set of issues consisting of price, delivery time and quantity. The top (dotted) graph represents the score obtained by the seller agent and the lower one the score obtained by the buyer agent. The agents have deadlines over which to reach an agreement. Buyer agents are designed to prefer low prices, short delivery times and low quantity. Sellers are designed to prefer high prices, long delivery times and high quantity.

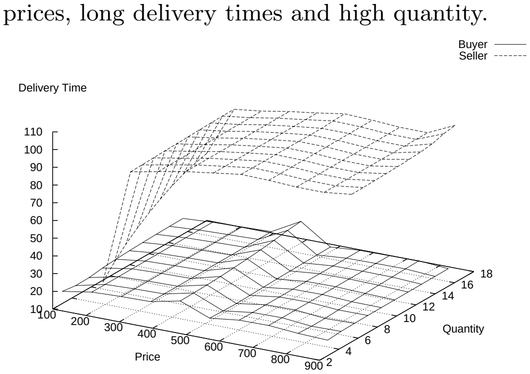

Figure 5: 30 Concurrent Buyer-Seller Negotiations

Reducing the bilateral negotiation mechanism to a *single* offer-accept or reject strategy, without prior suggestions and iterations, has the advantage of decreasing the costs by sending fewer messages or allowing agreements to be found without delay. The protocol allows only an accept or reject response to an offer. However assuming that deadlines are common knowledge is not practical. Moreover a single shot negotiation does not embody bargaining. Normally the two agents have different preferences and an intersection of their preferences is not guaranteed. The agents have no sure information about their opponents' strategies and preferences. In such cases, negotiation allows agents to probe each other's range of acceptability so that they may move their demands and issues set to create and fall within a mutual acceptance region. This process could take the form of a "trial and error" exchange of messages where the agents are providing feedback and learning what is acceptable. It is hard to see how a single offer negotiation allows the agents to revise their beliefs and preferences to converge towards an agreement. Moreover, proactive agents who negotiate in advance of their deadlines, so as to plan ahead or schedule their tasks, are discouraged by a reactive strategy. Also, in parallel negotiations, a seller or buyer may find other interesting deals rather than wait for a deadline to arrive.

5.2 Open Issues

Significant research has been carried out in AI regarding dominant strategies, Nash equilibria, Bayes-Nash equilibria, Bayesian equilibria for auctions, load balancing or resource sharing negotiation. How can we exploit such extensive work whilst integrating the different environmental strategies in section 4.1. In particular:

- Some approaches assume that the agents know or have agreed upon each other's strategies. If both agents agree on a conceding strategy, what is a true concession given that an agent may be using trade-offs – sending different issues set of the same value to itself, but different to others?

- Approaches focussing on one strategy with a single issue or action do not translate to our framework where there are multiple issues and bargaining with successive actions. We do not seek equilibrium for a take-it or leave-it offer, but rather the agents search each other's space for a mutually acceptable set of issues.

- The scoring function problems mentioned in section 4.3 remain when generating sets of issues. Also, how do the different environmental factors influence each issue, individually and in combination. For example, a low quality network may be offset by an agent having hours before its deadline and vice versa. An agent with seconds left may have enough time when the network is near-perfect. What is the influence of a good network and close deadline over a low priority negotiation? This raises the question of how to calculate the weights of the factors relative to each other.

- Complexity of analysing the strategy and finding the acceptable/optimum solutions rises exponentially with the number of issues, especially when the issues themselves have sub-attributes. For example, the quality of the network being parameterised with bandwidth, latency, etc.

- How is the quality of the network parameterised with its different sub-attributes? Ideally each network subfeature can influence the network and in turn the evaluation/generation strategies of an agent such as the rate of concession. Each feature would also influence choosing between the strategies and their weights, for example low bandwidth could imply attaching more importance to a time aware strategy $\mathcal{T}$ than to a strategy which computes the Pareto optimal solution or the Nash equilibrium.

6. NEGOTIATION MECHANISMS

The bilateral protocol in section 3 allows the participants to communicate with various speech-acts. In an alternating offers model, an agent has to choose between three decisions - accept, reject and counter-offer (or offer). In our framework, an agent can choose between more than these three actions and therefore there needs to be a strategy by which an agent may decide which speech-act is more profitable.

6.1 Levels of Commitment

We assign to the possible speech-acts levels of commitment. Commitment means that one agent binds itself to a potential contract while waiting for the other agent to either accept or reject its offer [12]. The latter paper proposes that protocols should have continuous levels of commitment based on a monetary penalty method, where a commitment break cost is assigned to each commitment strategy. The cost can change with time, the environment or be negotiated. In our case, commitments are not associated to only agreements but also at the various negotiation states. We do not consider rewards and punishments for respecting and breaking commitments since some states may be non-binding. The semantics of a speech-act allows to encode the associated range and level of commitment, for example as shown in table 1

Table 1: Levels of Commitment Associated with a Speech Act

Level	σ_1	σ_2	σ_3	σ_4
Speech-act	*request* *suggest*	*propose* *counter-propose*	*offer*	*agree, reject* *timeout*

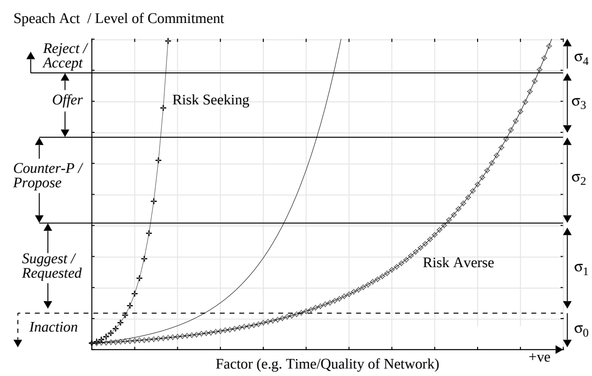

Figure 6: Commitment level of the Speech Acts

In figure 6, the x-axis represents the time an agent has left and the y-axis the levels of commitments depending on a speech-act. The curves indicate the risk behaviour of an agent. For the same amount of time left, two different levels of commitment can be chosen. Lower curves show a risk averse strategy while a higher growth rate shows a risk seeking behaviour. There remain a number of open issues when deciding upon a speech-act:

- The speech-act chosen may affect the strategy used for generating the issues set. For example, since a *request* cannot be agreed upon, an agent may be bold and make large concessions with a *suggestion* or a *request*, in order to investigate the acceptability region of a deal.

- Similarly, the generated or obtained set of issues may influence the speech-act chosen. For example, we allow an agent to accept an offer if what it received at time (*t-1*) is better than what it would have sent at time *t*.

- The different environmental factors and strategies determine the shape of the functions in figure 6. How are these functions derived from the environmental factors?

- Having different commitment levels may "give the game away". Sending an offer indicates that its deadline is nearer and its opponent just waits for the agent to concede.

We specify two algorithms when choosing the appropriate speech-act: 1) when an agent has received an acceptable issues set 2) when the received issues set is not acceptable. In the first case, the agent may accept if it is pressed for resources or otherwise further bargain to increase its gain. If the issues set is unacceptable, then the agent can only reject or make an offer if the resources are scarce. If resources are not scare, then an agent can search and bargain for a better set of issues than the received unacceptable one.

6.2 Choosing Levels of Commitment

In addition to simply mapping the different factors to levels of commitment, it is possible to specify each level at run-time. This is useful for coupling an agent's willingness to commit to its operational context (e.g. communication ability) and also allows the agent to deal with reputation and reliability information. For example, an agent should not be willing to increase its commitment levels for an agent that has previously reneged on its service provision agreements as compared to another that has not. Such discrimination may be provided by allowing an agent to perform a wholesale downward shift in its commitment boundaries (Figure 6) on, for example, a per negotiation partner basis. Alternatively, an agent that is encountering a general loss in communication reliability, may perform a temporary global downward shift (or totally redefine) the boundaries. This alleviates uncertainty associated with losing its ability to continue negotiating or its ability to communicate with and therefore invoke agreed remote services.

Rather than providing homogeneous degrees of commitment within each level, we further encode risk-seeking and averse behaviour within each level using a function that represents the certainty of the applicability of that level of commitment. For example, figure 7 shows five functions that represent an agent's (or user's) willingness to proceed (or revert) to various levels of commitment during negotiation. We expect that an agent's (or user'[s) willingness to change its level of commitment varies according to the associated risks of incorrectly under/overcommitting. For example, sending a *propose/counter-propose* action is not as risky as making an *offer* action since for the latter it is possible for a negotiation partner to prematurely conclude a negotiation sequence by sending an *agree* speech act. A *propose/counter-propose* action on the other hand does not allow this possibility, and moreover allows reversion to the less committed *requested* state. The functions may reflect this by having their maximal certainty either at the beginning or end of their defined regions (as in σ_2 and σ_3 respectively).

Allowing users and agents to define certainty values for levels of commitment at run-time can also be argued to act as a vehicle for restricting or granting agent flexibility. That is, by either removing its ability to act when operational factors enter certain regions. For example, either by allowing greater agent (and user) choice by overlapping the functions (see shaded region in figure 7), or conversly by associating undefined regions with a trigger to prompt users for a decisions.

As stated earlier, operational factors ($\mathcal{T}$,$\mathcal{Q}$,$\mathcal{E}$, $\mathcal{P}$ and $\mathcal{R}$) may be used in guiding negotiation strategy selection. Moreover, we believe that agents should take these factors into consideration when performing an evaluation of a received set of issues (and their value) and more importantly when deciding upon whether and when to respond. Figure 8 shows (simplified) buyer agent expectations about the likely region of acceptable deals (as determined from its own knowledge of acceptable deals exhibited by a supplier agent). The distance between the current issue valuation of the customer and that of the supplier's last offer may be used to guide the negotiation strategy and response generation function in order to reduce the number of message exchanges. However, timeout/ deadline information (and other operational factors) need to be convolved with the raw distance.

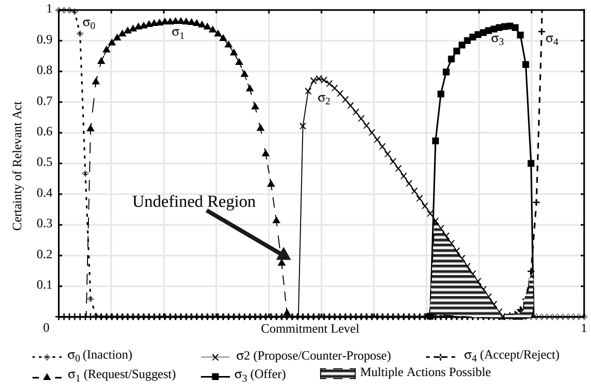

Figure 7: Commitment Level Functions

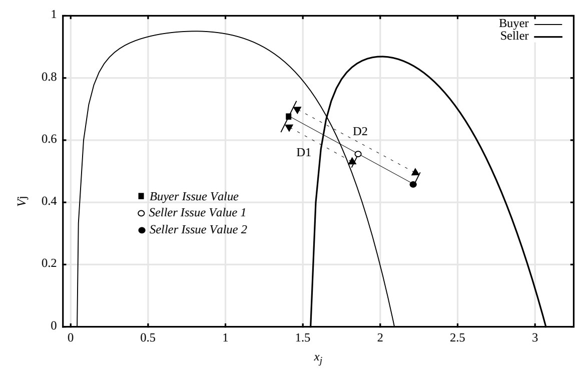

Figure 8: Non-Trivial Distance Metrics

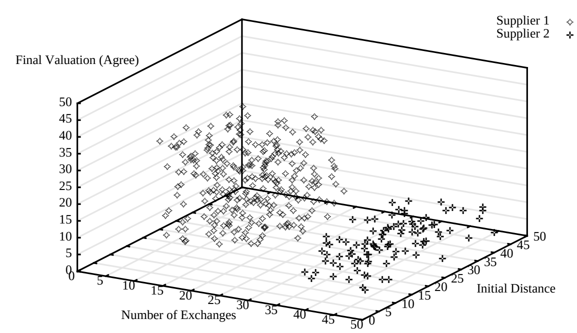

Figure 9: Succesful Past Negotiations

Figure 9 shows a customer agents experience of the final value and number of exchanges (implying processing/messaging costs) as related to the initial distance between the first issue/value sets exchanged. Supplier 2 exhibits a tendency to require longer exchanges that supplier 1 and also has results in lower quality deals. It would appear sensible for the customer to approach supplier 1 instead of 2. However, this does not take into account the operational factors mentioned above. Viz., that distance *per se* needs to be convoled with other run-time information such as individual supplier timeouts.

Figure 10 portrays an agent's free communicative resources for some period into the future as derived from existing commitments and expectations of resource usage of other negotiation threads. Three regions are defined: 1) (LHS) the current local environment of the agent (e.g. a desk-top PC with high-speed LAN), 2) (MID) a migration phase where the agent moves from the former device to remote device,

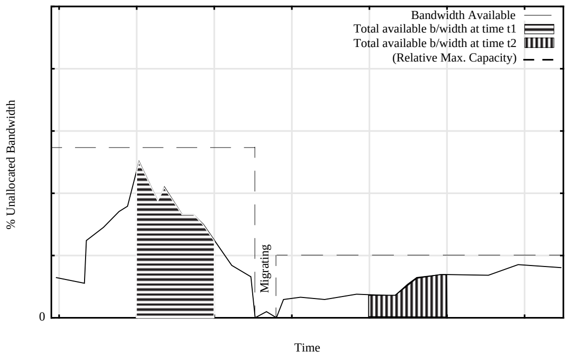

Figure 10: Projected Future Resource Availability

and 3) (RHS) communicative resources commited/available within the remote device. When determining which supplier to negotiate with, the agent must include the following points in its decision: a) the locations of supplier 1 and 2, b) whether distance is reduced to a cost that is relative to the location context of the agent or an absolute comparison, and c) if timeouts and deadlines allow, whether negotiation can be deferred to a later time or brough forward to take advantange of current resource availability. Specifically, we do not assume that suppliers have the same timeouts and therefore simple comparison of distances (e.g. D1 and D2 on figure 8) can extend over the migration boundary resulting in non-trivial mapping between distance, expected exchanges and actual resource costs.

Lastly allowance should also be made for the need for predictability to allow an agent to plan more effectively and reason more accurately about expected resource costs. A risk averse agent may nevertheless choose to select supplier 2 since the region defining the costs associated with negotiating (in fig 9) is less spread out than that for supplier 1. That is, that the customer may choose supplier 2 since it can be more certain about the costs it will incur during the negotiation. The customer may therefore trade utility for predictability thereby allowing it to plan its future resource commitments (e.g. other negotiations) more effectively.

7. CONCLUSIONS

This paper discusses negotiation mechanisms for m-services trading as an outlook on the future of electronic commerce via agents located on mobile devices. In this domain, the characteristics of wireless communications constrain the interactions between agents in an open market, in addition to other factors such as resources, time and experience guide the negotiation strategies. We discuss the facilitation of peer-to-peer bilateral negotiation mechanisms and we focus on the trading of services in an m-commerce environment rather than the sharing of a resource. We also mention the open issues we encountered whilst designing and implementing negotiation strategies in such environments.

Other open issues that represent avenues for future work include dealing with timeouts occuring for example when an agent does not respond because it has crashed or it does not find it worthwhile to do so, or the network has failed. Ascertaining whether an interaction has timedout could end up with infinite acknowledgements or negative acknowledgements. On the other hand, an agent may not have any incentive to send negative replies because it is not interested in a service.

Most of existing work on bargaining strategies apply to a single negotiation process or computing one response, and do not consider the possibility of concurrent negotiation threads (apart from double auctions). Parallel and concurrent negotiations affect each other – their strategies and decisions are affected by the other threads. For example, an agent performing 50 negotiations requires more bandwidth than when engaged in a single negotiation. Finally, there are open issues for agents operating in supply chains. For example, an agent may be simultaneously negotiating to sell and item that it is also currently negotiating to purchase.

8. ACKNOWLEDGEMENTS

The work reported in this paper has formed part of the Software Based Systems area of the Core 2 Research Programme of the Virtual Centre of Excellence in Mobile and Personal Communications, Mobile VCE, (www.mobilevce.com) whose funding support, including that of EPSRC, is gratefully acknowledged. More detailed technical reports on this research are available to Industrial Members of Mobile VCE.

9. REFERENCES

[1] D. Chalmers and M. Sloman. A survey of quality of service in mobile computing environments. *IEEE Communications Surveys*, 2(2), 1999.

[2] P. Faratin, C. Sierra, and N. R. Jennings. Using similarity criteria to make negotiation trade-offs. In *ICMAS 2000*, pages 119–126, 2000.

[3] N. R. Jennings, P. Faratin, L. A., S. Parsons, C. Sierra, and M. Wooldridge. Automated negotiation: prospects, methods and challenges. *International Journal of Group Decision and Negotiation*, 10(2):199–215, 2001.

[4] S. Kraus. Beliefs, time and incomplete information in multiple encounter negotiations among autonomous agents. *Annals Math. and AI*, 20(1-4):111–159, 1996.

[5] S. Kraus, J. Wilkenfeld, and G. Zlotkin. Multiagent negotiation under time constraints. *AI*, 75(2):297–345, 1995.

[6] K. Larson and T. Sandholm. An alternating offers bargaining model for computationally limited agents. In *Proceedings of AAMAS'02*, 2002.

[7] J. Odell, H. Parunak, and B. Bauer. Representing Agent Interaction Protocols in UML. In *AOSE Workshop*, pages 121–140, 2001.

[8] S. Paurobally. *Rational Agents and the Processes and States of Negotiation*. PhD thesis, Imperial College, 2002.

[9] H. Raiffa. *The art and science of negotiation*. Harvard University Press, 1982.

[10] J. Rosenschein and G. Zlotkin. *Rules of Encounter: Designing Conventions for Automated Negotiation among Computers*. MIT Press, 1994.

[11] N. Sadeh. *M-Commerce: Technologies, Services, and Business Models*. Wiley, USA, 2002.

[12] T. Sandholm and V. Lesser. Issues in automated negotiation and electronic commerce: Extending the contract net framework. In *ICMAS '95*, pages 328–335, 1995.

[13] T. Sandholm and N. Vulkan. Bargaining with deadlines. In *AAAI*, pages 44–51, 1999.

[14] P. J. Turner and al. Scenarios for future communications environments. In *Technical Report ECSTR-IAM02-005, ECS Dept., Southampton University, UK. 2002.*

[15] M. Wooldridge. *An introduction to multiagent systems*. John Wiley and Sons Ltd., 2002.