Available online at www.sciencedirect.com

SCIENCE DIRECT

Knowledge-Based Systems 17 (2004) 103–111

Knowledge-Based SYSTEMS

www.elsevier.com/locate/knosys

Agent-based formation of virtual organisations

Timothy J. Norman[a,*], Alun Preece[a], Stuart Chalmers[a], Nicholas R. Jennings[b], Michael Luck[b], Viet D. Dang[b], Thuc D. Nguyen[b], Vikas Deora[c], Jianhua Shao[c], W. Alex Gray[c], Nick J. Fiddian[c]

[a] *Department Computing Science, University of Aberdeen, Aberdeen, UK*
[b] *School of Electronics and Computing Science, University of Southampton, Southampton, UK*
[c] *Department Computer Science, Cardiff University, Cardiff, UK*

Available online 12 April 2004

Abstract

Virtual organisations (VOs) are composed of a number of individuals, departments or organisations each of which has a range of capabilities and resources at their disposal. These VOs are formed so that resources may be pooled and services combined with a view to exploiting a perceived market niche. However, in the modern commercial environment it is essential to respond rapidly to changes in the market to remain competitive. Thus, there is a need for robust, agile, flexible systems to support the process of VO management. Within the CONOISE (www.conoise.org) project, agent-based models and techniques are being developed for the automated formation and maintenance of virtual organisations. In this paper we focus on the former, namely how an effective VO may be formed rapidly for a specified purpose.

Keywords: Virtual organisations; Agents

1. Introduction

Virtual organisations (VOs) are composed of a number of semi-independent autonomous entities (representing different individuals, departments and organisations) each of which has a range of problem solving capabilities and resources at their disposal. These entities co-exist and sometimes compete with one another in a ubiquitous virtual marketplace. Each entity attempts to attract the attention of potential customers by describing the cost and qualities of its services, with the goal of selling them in a way that maximises their individual gain. Sometimes, however, one or more of the entities may realise there are potential benefits to be obtained from pooling resources: either with a competitor (to form a coalition) or with an entity with complementary expertise (to offer a new type of service). When this potential is recognised, the relevant entities go through a process of trying to form a new VO to exploit the perceived niche. Consider two examples. First, suppose that two relatively small airline companies with complementary routes agree to cooperate and coordinate their services so that they may offer flights, as a coalition, between a wider range of destinations, with a view to becoming more competitive in this market. Second, a streamed video content provider and a high bandwidth mobile service provider may agree to collaborate in the delivery of such content as a service to mobile devices (this corresponds to a new type of service). Given the independent nature of the entities involved, there are numerous reasons why the formation of a VO may fail. If it succeeds, however, the collection of independent entities will act as a single conceptual unit in the context of the proposed service (they may continue to retain their individual identity outside this context). In particular, the participants must cooperate and coordinate their activities in delivering the services of this newly formed organisation—the participants must have the ability to manage the VO effectively. In dynamic environments, however, the context may change at any time, such that the VO is no longer viable. It will then need to either disband or re-arrange itself into a new organisation that better fits the prevailing circumstances.

This automated formation and ongoing management of virtual organisations in open environments represents a major research challenge. A key objective in putting such organisations together is to ensure that they are both agile (able to adapt to changing circumstances) and resilient (able to achieve their objectives in a dynamic and uncertain environment). In such environments, the participants' behaviour will be informed by exploiting a number of

[*] Corresponding author.
E-mail address: tnorman@csd.abdn.ac.uk (T.J. Norman).

0950-7051/$ - see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.knosys.2004.03.005

diverse forms of information—advertisements (capabilities and reputations of individual agents), meta-data (schemas and ontologies) and information resources (databases and knowledge bases).

The novel contribution of the CONOISE project is to provide a model of VO management that operates in a robust and resilient manner in complex electronic commerce scenarios. In particular, we focus on the first element of a complete VO management system: VO formation. The formation of a virtual organisation within the CONOISE system is grounded on three key technologies: the decision-making mechanism of an individual agent, an auction mechanism for the allocation of contracts, and the representation of services. The contribution of this paper lies in the integration of these technologies to provide a solution to the problem of forming effective virtual organisations in complex, information rich environments.

Before the CONOISE solution to VO formation is discussed in detail (Section 3), it is important to have a better understanding of the issues that must be considered in developing a computational model of VO formation and to present a specific scenario in which the ideas presented in this paper may be grounded (Section 2). Following the detail on VO formation, we discuss avenues for future development by returning to the example introduced in Section 2 and present our conclusions to this paper (Section 6).

2. A VO formation scenario

In presenting an overall picture of the CONOISE VO management process, we will use a specific scenario. This scenario illustrates a number of important characteristics that must be taken into account in the development of an effective VO management system. First, there may be multiple services available from a number of agents representing independent organisations. Multiple agents may offer broadly similar services. The services themselves are described by multiple attributes; for example, price, quality, and delivery time. The services available may change over time: new services may become available, or agents may alter the way in which existing services are offered. Services may differ in terms of the number and heterogeneity of the tasks involved in the delivery of the service and their degree of interdependence, and the type and frequency of interactions between different customers while the service is being delivered. The agents involved in the system may also employ different policies for dealing with the uncertainty inherent in such a domain; for example, an agent may generate slack resources to limit the possibility of a loss in service to the customer, or it may employ rigorous coordination mechanisms to improve supply chain integration.

With these issues in mind, consider the following scenario. A user wants to purchase and receive a monthly movie subscription package on his PDA/phone, and a monthly news service. The user also wants a monthly package for his PDA/phone that includes 30 free text messages and at least 50 free minutes per month. This is a reasonably complex and realistic set of requirements that incorporates four types of service: movies, news, text messaging and a phone service. Within the scenario, a requirements agent (RA), represents this user. In addition to the agent representing the customer's requirements, there are a number of agents representing service providers (SP1–SPn). The services that these agents provide are captured as 'packages', which may represent quite complex offers (see Section 3.2). Suppose that agent SP1 offers a number of packages containing news and movies services. The packages on offer may include, for example, news and movies services for one month at £30 per month, and the same service for six months at £25 per month.

Prior to the RA initiating the process of VO formation, it is assumed that each service provider advertises the services that they offer—e.g. movies or text messaging—to a yellow pages agent (YP). This agent is consulted by the RA and asked to recommend agents that have advertised the ability to deliver movies, news, text messaging or phone services. Following the receipt of this information, the RA will distribute a call for bids to fulfill a specific set of requirements (see Fig. 1).

In this call for proposals the units—movies, news, text messaging and phone—and the values associated will represent components in a package and the values and attributes of that package. The service provider agents must now decide whether and what to bid in response to this call. Suppose that there are four service provider agents contacted in this way—SP1–SP4—and the packages on offer are those illustrated in Table 1. Note that SP3 imposes a further constraint on the package that it offers: both the services stated in the package must be taken together. How these packages are constructed is not specified, but an individual service provider could have put a package together from its own resources or through the formation of a virtual organisation.

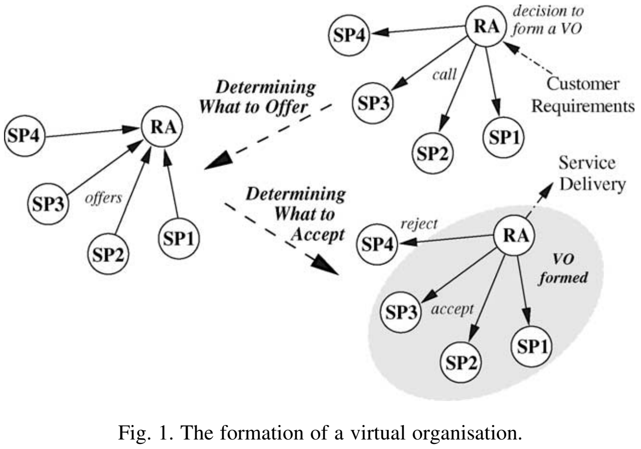

Fig. 1. The formation of a virtual organisation.

Table 1
An example set of available packages

Service provider	Movies (per month)	News (no. of daily updates)	Text (no. of free messages)	Phone (no. of free min.)
SP1	10	24		
SP2		72		
SP3			120	30
SP4	5			30

The RA must, once the deadline for proposals to be submitted has passed, select some combination of services that best suits the needs of the user. An appropriate combination of services given these bids is to take the movies service offered by SP1 (note that this package may be split into its component services), the news service offered by SP2 and both text and phone services offered by SP3. Although the phone service requirement is not met, this represents the best choice given the circumstances. Thus, once proposal acceptances and rejections are sent to the agents that submitted bids, a virtual organisation is formed that involves RA, SP1, SP2 and SP3.

We will return to this scenario throughout Section 3 and then again in Section 6 where VO maintenance is discussed as the principal focus of future development. However, at this point we present the detail of the CONOISE VO formation mechanism.

3. The formation of a virtual organisation

As discussed in Section 1, the novelty of this research lies in the technologies being employed in the management of virtual organisations and their integration in a coherent VO management system. Here we focus on the first element of this integrated system: the formation of a VO. In developing a model of VO formation, there are a number of issues that must be taken into account including:

- An agent that is considering whether to offer to join a VO must determine the conditions under which it is profitable for it to join (see Section 3.1).
- An agent must be able to recognise circumstances in which it should initiate VO formation (see Section 3.1).
- The agent that initiates the VO formation process must, given a number of offers, determine the best combination of business partners (see Section 3.2).
- In the support of these decisions, rich descriptions of service quality are required to capture the extent to which services meet the expectations of consumers (see Section 3.3).

3.1. Determining what to offer

The purpose of a service provider agent is to be able to create a bid in reply to a call for services, and decide how much resource it can, and more importantly, how much resource it wants to provide as a bid for the procurement of that service. Furthermore, any agent may, when considering what to offer, take on the role of the RA in Fig. 1 and issue a call for bids if it identifies a shortfall in its existing resources available. Each agent must, therefore, be able to act as a contractor and supplier in any given situation.

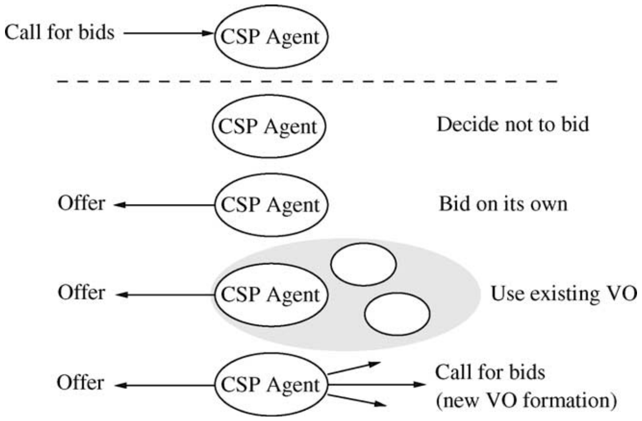

Fig. 2. The agent decision making process.

To give such dual-purpose functionality, we have designed a Constraint Satisfaction Program (CSP) that models the decision making process the agent must take in such scenarios.

Fig. 2 shows one such scenario, where the agent acts as the supplier and receives a call for bids. It has the following possible responses: (i) It can decide not to bid for the service; (ii) It can bid using just its own resources; (iii) It can provide a bid from within an existing VO collaboration utilising the combined VO's resources; or (iv) It identifies a need for extra resources not available within the existing VO. We can see that the last option represents the scenario where the agent becomes the contractor, and itself begins the process of issuing a call for bids to other agents in the environment.

The technique used to provide the decision making process is based on a cumulative scheduling CSP [6]. Usually, this is defined as the maximum allowable limit from a finite 'pool' of resource that can be used collectively by the agents at any given time [1]. We define our problem differently; rather than the agents taking resources from a communal resource, we have the agents contributing to the communal pool, and we define a minimum allowable limit so that the set of agents must provide this service at least or above the required threshold limit over the required time. If it is not possible, then we use the CSP to highlight the deficit and can then look to contracting-out for the provision of this shortfall.

To explain our cumulative scheduling based algorithm, we first define the problem. Given a set of n agents in a VO, each of whom can provide a specific finite amount of a resource R, $\{R_1 \ldots R_n\}$, a set of start times describing when

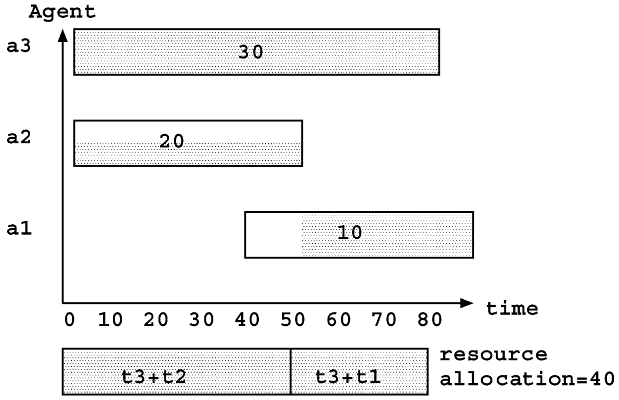

Fig. 3. An example schedule.

the agent can begin providing each of the resources $\{S_1 \ldots S_n\}$ and a set of durations over which the resource is available $\{D_1 \ldots D_n\}$ we can say, for an agent $i \in \{1 \ldots n\}$, that the function $\delta_i(t)$ evaluates to 1 if the current time t is within the agent's resource start and end time $(S_i < t \leq (S_i + D_i))$, and 0 otherwise. Then, an amount r of resource R is available over a time period $1 \ldots v$ iff$\forall t \in \{1 \ldots v\}\left(\sum_{i=1}^{n} R_i \delta_i(t)\right) \geq r$. In other words, the total sum of the resource provided by the set of agents with indices $\{1 \ldots n\}$ in a VO at any time between $1 \ldots t$ does not fall below the resource limit r specified. Using this representation means that we can also use constraints on the agent resource domains to represent existing commitments on those resources.

In our scenario, this helps us to model the decision making process as the agent can look at the existing partners in its VO, as well as its own resources and the existing commitments, and see whether it can accommodate the new allocation of resources asked of it. As an example, let us look at an agent a1 who is in a VO with two other agents a2, a3. All can provide a certain amount of bandwidth (10, 20 and 30 units, respectively). Agent a1 is asked to provide a total bandwidth amount of 40 units (as described in Section 1) from time 0 to 80, so it uses the knowledge of the amount of resources contributed from the other agents in the VO (along with its own) to work out if this is possible. Fig. 3 shows an example allocation. A total rate of 40 units is provided by a3 and a2 between 0 and 50, then by a3 and a1 between 50 and 80. We can also add constraints on the resources available for each agent at each point in time to represent commitments under other contracts.

Of course there are many permutations that we can have in this resource allocation process. What we have described so far shows what the agent can do, but we also want to be able to model a utility that allows the agent to choose between competing viable allocations (i.e. decide what it wants to do).

We have implemented this utility using constraint reification, where each constraint on the domain of the resource has an associated value, 1 or 0, which depends on the success or failure of the constraint. For instance, using SICStus Prolog[1] notation, $\mathsf{X} < \mathsf{Y} \neq \ \Leftrightarrow \mathsf{B}$ states that if X is less than Y, the variable B is set to 1, otherwise it is set to 0. When the agents try to provide a new resource we take into account the current commitments of the agents (all the constraints currently posted against the resources) and we get a set of reified values for each commitment which we can then use to see which constraints are satisfiable alongside the new call for bids, and which ones 'fail', and so have a 0 value in their reification, that is, the resources cannot be allocated in the current situation. We can also highlight where the new bid is failing and identify the shortfall. Using this information, we also have a basis on which we can look at quality and pricing metrics (see Section 3.3) for commitments in comparison to the new resource being bid for, and this therefore allows us to prioritise the commitments we have against any new ones that might arise. Before we discuss quality issues, however, we will address the problem of which offers the agent initiating VO formation should accept to create the best, or at least a satisfactory, VO.

3.2. Determining what to accept

Since VOs do not have a rigid organisational framework, the selection of partners is one of the most important activities in the formation of the VO ([16]). However, there are several requirements that need to be met by this process:

- The most suitable set of partners from those that are available should be selected. In this context, most suitable means the ones with lowest price bids. Note that the *price* here does not just mean the monetary value of the bids but may be a combined rating value, calculated from monetary value and other attributes of the goods/services offered by the partners (e.g. time).
- The selection should occur within a computationally reasonable time frame so that the market niche can be exploited as it becomes available.
- The potential partners should be able to vary their bid depending on their involvement in the VO. Thus, for example, a partner may be willing to complete services more cheaply if it has a high degree of involvement in the VO (because the intrinsic costs can be depreciated over many instances). In contrast, if a partner has a comparatively small involvement then the unit cost may be much higher.

Given the open nature of the environment and the lack of a pre-ordained structure, we believe this creation process is best achieved using some form of marketplace structure (auction). This is because markets are a highly effective structure for allocating resources in situations in which there are many self-interested and autonomous stake-holders.

[1] The cumulative scheduling algorithm is implemented using the finite domain constraint library in SICStus.

There are, however, many different types of auction (see Ref. [24] for a classification) but in this work it was decided to adopt a combinatorial auction approach. A combinatorial auction is a sophisticated type of auction where multiple units of multiple (potentially inter-related) items are traded simultaneously. In a combinatorial auction, bidders may bid for arbitrary combinations of items. For example, a single bid may be for 5 movies, 24 news updates (per day) and 20 min of phone at a total price p per month. A more complicated bid may be for q_1 movies and q_2 news updates at price $(30 \times q_1 + 3 \times q_2)$ if $q_1 < 10$ or $q_2 < 24$, and at price $20 \times q_1 + 2 \times q_2$ if $q_1 \geq 10$ and $q_2 \geq 24$. This particular type of auction is suitable for this problem because the degree of flexibility in expressing offers allows the potential partners to vary their bid depending on their involvement in the VO. However, the main disadvantages of combinatorial auctions stem from the lack of a compact and expressive bid representation and efficient clearing algorithms for determining the prices, quantities and trading partners as a function of the bids made. Without such algorithms, because of the computational complexity of the problem, there may be unacceptable delays for auctions that have only a medium number of participants. Thus, in the CONOISE context, a compact and expressive bid representation language and efficient clearing algorithms for combinatorial auctions have been developed [8].

Specifically, we developed a bid presentation language where the price of a package, $P_i(r_1, \ldots, r_m)$ is specified as: $\omega_i(t_1, \ldots, t_m)\left(\sum_{j=1}^{m} P_i j(r_j)\right)$, where P_i^j is the price function of agent i for item j, in the form of a piecewise linear curve (i.e function's graph is composed of many segments, each of which is linear), t_j is the segment number of P_i^j that r_j belongs to and ω_i is a function that expresses correlations between items in the set of segments.

More precisely, each piece-wise linear function P_i^j is composed of N_i^j linear segments, numbered from 1 to N_i^j. Each individual segment with segment number $l, 1 \leq l \leq N_i^j$, is described by a starting quantity $s_{i,l}^j$ and an ending quantity $e_{i,l}^j$, a unit price $\pi_{i,l}^j$ and a fixed price $c_{i,l}^j$, with the meaning that: bidder i wants to trade any r units of item $j, s_{i,l}^j \leq r \leq e_{i,l}^j$ with the price $P = \pi_{i,l}^j r + c_{i,l}^j$.

Note that the segments are not required to be continuous; that is, $(s_{i,l+1}^j - e_{i,l}^j)$ may not equal 1. Also, for convenience, we call segment number 0 the segment in which the starting quantity, the ending quantity, the unit price and the fixed price are all equal to 0. Thus, the number of segments of P_i^j, including this special segment, will equal $N_i^j + 1$.

The correlation function ω_i has many potential uses in real-life scenarios. For example, suppose bidder i, selling three items (movies, text messages and phone calls), wants to express things like "I am willing to sell 100 min of phone calls per month and 50 text messages per month together with a price P, but not separately". Using our correlation function, this can be expressed by adding segments t_1 and t_2, which contain only 100 and 50, to the functions P_i^1 and P_i^2, respectively, then giving $\omega_i(t_1, t_2, t_3)$ a very small value, for every t_3, and giving $P_i^1(100)$ and $P_i^2(50)$ very big values. This way, the auctioneer will never choose to buy 100 min of phone calls or 50 text messages separately.

This means of representing bids is novel and superior to popular bid representations. Compared with other work in this area [9,19] it is more expressive as it allows bidders to detail the correlation between separate items. Compared to XOR atomic proposition presentations, it is nearly as expressive but much more compact. Moreover, this case is important to consider because piecewise linear curves are commonly used in industrial trading applications [9] and any general curve can be approximated arbitrarily closely by a family of such functions [19].

Two sets of clearing algorithms have been developed: one with polynomial complexity and has been shown to produce a solution that is within a finite bound of the optimal [8], while the other is not polynomial but is guaranteed to produce the optimal allocation [7]. In particular, the former uses a greedy approach, and has a running time of $O(n2)$, where n is the number of bidders. The solution it produces is shown to be within a finite bound of the optimal, which is proportional to n and K^{m-1}, where m is the number of items and K is a small constant. On the other hand, the latter is guaranteed to produce the optimal allocation, and has a worst-case running time that is proportional to $mn(K' + 1)^{mn}$, where K' is the upper bound on the number of segments of P_i^j. As these two sets of algorithms provide a trade-off between running time and optimality of solution, they provide the user with more flexibility. In cases where the running time is more crucial, the polynomial algorithms would be more appropriate, while in cases where optimality of the solution is more desirable, the optimal algorithms will be better suited.

3.3. Managing quality of delivery

In this section we describe the role of the Quality Agent (QA) in the CONOISE solution to the problem of VO management. QA is responsible for collecting information related to the quality of the services offered by SPs, and to supply this information to RA for it to use in the process of forming a VO. The information about Quality of Service (QoS) provides another basis for negotiation (in addition to the price), and thus is important to the process of VO formation.

There exist various interpretations of QoS in the literature and a large number of methods have been proposed for managing QoS in marketing, e-commerce and other systems [2,13]. While some qualities, such as network traffic and speed, may be monitored automatically, more subjective qualities, for example, frequency of news updates, require user preference information. Existing methods typically invite users to rate a service in absolute terms, e.g. good, bad or 7 out of 10. Such quality ratings may not be very meaningful or can even be misleading in some cases, because the context within which the ratings are derived is not known.

In CONOISE, we attempt to address the problem by introducing a model for collecting and monitoring QoS relative to specific users or types of user. That is, we attempt to collect from service users (or their agents) QoS ratings as well as their expectations on QoS, so that we can measure how well a delivered service meets users' expectations. More specifically, let S be a service and $q_1, q_2, \ldots, q_n$ be a set of attributes with which we wish to monitor QoS for S. We collect the following from service users for each q_i of $S : \langle QE(q_i), QP(q_i), QR(q_i)\rangle$ where $QE(q_i)$ represents the QoS that the user expects from S with respect to q_i, $QP(q_i)$ the actual QoS of q_i perceived or experienced by the user after using S, and $QR(q_i)$ the rating that the user gives to S in terms of q_i. All three values are represented by real numbers in the range [0,1]. For example, the quality of news update frequency may be rated by a user as $\langle QE(\text{fr}) = 0.65, QP(\text{fr}) = 0.76, QR(\text{fr}) = 0.85\rangle$ indicating that an above average frequency was expected (0.65), the actual update delivery was more frequent (0.76) and, consequently, the quality of service was considered to be good (0.85).

To combine QoS ratings collected from service users into an overall assessment of quality for a given service S, we perform two calculations: (i) combining individual ratings for each q_i of S into an aggregate rating, and (ii) combining the ratings for individual q_i's into an overall rating for S. Currently, we treat all quality attributes to be of equal importance and (ii) is derived by a simple average of the individual ratings. But it is possible to consider a weighted average so that the fact that some attributes are more significant than the others may be taken into account. The combination of individual ratings depends on the quality assessment request, R, received by the QA. If R specifies no quality expectation on q_i, then $Q(q_i) = \sum_{j=1}^{k} w_j \times QR_j(q_i)$. This is equivalent to the majority of existing approaches to quality calculation; the overall rating for q_i is a weighted sum of individual ratings, and the weights are used to allow factors such as trust to be taken into account [25]. If R specifies a quality expectation $E(q_i) = \alpha \in [0,1]$ on q_i : (the quality expectation on q_i is α), then $Q(q_i) = \sum_{j=1}^{m} w_j \times QR'_j(q_i)$ Here, $QR'_j(q_i)$ is a rating whose corresponding expectation $QE'_j(q_i)$ is compatible with $E(q_i) = \alpha$. In this paper, we use a simple criterion for determining whether the two are compatible: $QE'_j(q_i)$ and $E(q_i) = \alpha$ are compatible if $|QE'_j(q_i) - \alpha| \leq \delta$, where δ is a constant. However, more complex forms of compatibility test are possible, for example, by specifying quality expectations as ranges and by allowing fuzzy matching between $QE'_j(q_i)$ and $E(q_i) = \alpha$. Further discussion on these issues is beyond the scope of this paper.

We now illustrate our quality model and assessment by considering the scenario given in Section 2. Suppose that we have six agents (A1–A6) who have used the news services provided by SP1 and SP2. Each agent is then asked to rate the services in terms of news update frequency. Table 2 shows the ratings collected.

Table 2
A set of example quality ratings collected for SP1 and SP2

Agent	SP1	SP2
A1	⟨0.9, 0.7, 0.5⟩	
A2	⟨0.4, 0.4, 0.6⟩	⟨0.4, 0.5, 0.9⟩
A3	⟨0.8, 0.6, 0.3⟩	
A4		⟨0.4, 0.5, 0.8⟩
A5	⟨0.9, 0.7, 0.5⟩	
A6	⟨0.9, 0.7, 0.6⟩	⟨0.9, 0.4, 0.2⟩

In this example, the users of SP1 have high expectations, but do not receive what they expect. Users of SP2, on the other hand, do not have high expectations but are generally satisfied with the service. It is this difference in expectation that QA exploits in assessing QoS for services. Suppose that QA is asked to assess QoS for SP1 and SP2 in terms of news update frequency (fr), given $E(\text{fr})$ not specified, $E(\text{fr}) = 0.5$ and $E(\text{fr}) = 0.8$, respectively. Assuming that we have $\delta = 0.1$, the result of calculation is: (i) when $E(\text{fr})$ not specified, QoS of SP1 is 0.50 and QoS of SP2 is 0.63; (ii) when $E(\text{fr}) = 0.5$, QoS of SP1 is 0.60 and QoS of SP2 is 0.85; and (iii) when $E(\text{fr}) = 0.8$, QoS of SP1 is 0.50 and QoS of SP2 is 0.20. The quality ratings for SP1 and SP2 can, therefore, vary with respect to expectation. This is in contrast to more conventional approaches to quality calculation that do not consider user expectations (equivalent to $E(q_i)$ not specified), our method gives a more meaningful rating for a service on a case-by-case basis.

Finally, it is worth mentioning that although $QP(q_i)$, the quality perceived by the user, is not used in quality calculation at the moment, it can play an important role in deriving more accurate quality assessments. For example, by monitoring the relationship between $QR(q_i)$ and $|QE(q_i) - QP(q_i)|$ over a period of time with sufficient rating data, we can determine whether a particular agent has been 'harsh' in rating services. By factoring such knowledge into quality calculations, we can deliver more accurate QoS assessment for the RA agent.

4. VO management and future research

In this paper, we have focussed our attention on VO formation. However, once formed, a VO must be managed effectively, and, possibly, restructured if new opportunities are identified or problems encountered. Returning to the scenario introduced in Section 2, suppose that a new service provider, SP5, enters the environment offering a text messaging service with 200 free messages per month. This opportunity may have been recognised by the RA while monitoring new package advertisements, or by SP5 approaching the manager of the existing VO. If such an opportunity is recognised by RA it may consider re-negotiating the contracts that bind this VO together.

Suppose that RA attempts to re-negotiate with SP3 for just phone calls, and take the text messaging service provided by SP5. However, SP3's deal has a constraint that says both phone calls and text-messaging services must be taken together as a package. RA may then decide to seek an alternative provider of phone calls (in this case SP4). (There may, of course, be penalties to pay for withdrawing from the contract with SP3, and such factors must be taken into account when RA considers restructuring the VO). As a result of this restructuring, SP3 ceases to be a member of the VO, but two new suppliers—SP4 and SP5—become involved. It is not only opportunities in the form of new service packages becoming available that the manager of a VO (in this example RA) must take into account; problems may occur that force the restructuring of the VO; for example, SP2 may withdraw its news service.

During the life time of a VO, automated negotiation[2] can be used to maintain or extend its formation. Consider two possible situations: when a VO (composed of n agents $\{A_1, A_2 \ldots A_n\}$) has been formed, one agent A_i drops out due to a specific reason (e.g. communication failure or it is no longer in its own self interest to be involved). In this case, the current VO should not be dissolved because the remaining agents are still committed to their aims and objectives. Therefore, another agent should be summoned to replace A_i. In this situation, RA has to find the new agent within minimum time and cost. The second situation is when a VO has been formed and is operating and a new requirement is introduced that the current VO is not capable of handling. In order to enhance the current functionality of the VO, one or more agents need to be added to the formation. Again, this change to the formation of the VO is carried out via the process of negotiation.

In more detail, in the CONOISE context, when RA needs to find a particular agent for a specific requirement, it first requests the list of capable agents from the yellow pages agent (YP). From this list, RA then negotiates with each of the agents in order to find the most suitable candidate. To minimise the time spent on this process, we decided to develop a negotiation model that permits multiple concurrent negotiations [15]. This concurrency also enables the agent to examine more potential solutions in a given time period and we have shown empirically, that this leads to better deals than either single partner negotiations [14].

Specifically, this negotiation model adopts a heuristic approach in which negotiation behaviour is determined by a number of tactics determined by different environment factors and by a strategy that realizes this importance of the different tactics (cf. Ref. [10]). The model considers the situation in which there is one agent (called the buyer) trying to negotiate a service with a number of other agents (called the sellers). The buyer uses a number of concurrent threads, each of which negotiates with a specific seller using a specific strategy. During the negotiation, the buyer tries to categorise the type of seller it is dealing with. Based on the feedback from these threads, the buyer may change its negotiation strategy for a thread, according to the type of the corresponding seller. When all the negotiations terminate, the buyer selects the seller that has produced the highest agreement value as the one that it actually confirms the deal with.

Turning now to future research, an area of particular interest in this project is that of trust and reputation. Whenever interactions take place between different agents, trust in and reputation of agent are significant, especially in the context of virtual organisations in which agents must rely on each other to ensure coherent and effective behaviour. Though some work has been done in this area, the focus on VOs has been limited, with the majority adopting the stance of assuming complete trust, and avoiding the issue. However, as discussed by Luck et al. [12], questions of deception and fraud in communication and interaction, of assurance and reputation, and of risk and confidence, are critical, especially where interactions take place with new partners. In future work, we will seek to understand the requirements for trust and reputation and evaluate existing models with regard to identifying the specific needs of VOs. Among the potential mechanisms for trust and reputation are centralised reputation systems currently used in the context of marketplaces, and personalised reputation systems in social networks, both of which will be explored.

5. Related work

Because of the potential economic benefits of VOs, there is starting to be considerable research in this area [4,5,20,21]. For example, the NIIIP (National Industrial Information Infrastructure Protocols),[3] Production Planning and Management in an Extended Enterprise (PRODNET) [3] and Virtual Enterprise Generic Applications (VEGA*) [23] projects are concerned with the development of IT and/or cooperation platforms for VOs. The NIIIP project aims to build an information infrastructure, which supports the whole virtual enterprise life-cycle. Specifically, it aims to provide technical foundations for the implementation of virtual enterprises; to establish an open, standards-based software infrastructure protocol that integrates heterogeneous and distributed processes, data, and computing environments across the US manufacturing base; to implement NIIIP from emerging, existing, and de facto standards and system technologies; and to accelerate consensus on standards that promote the deployment of VEs (NIIIP). The other projects have smaller scopes. Specifically, the VEGA project aims to establish an information infrastructure to support

[2] A process by which a joint decision is reached by two or more agents.

[3] http://www.niiip.org/.

the technical and business operations of VEs using groupware tools and distributed architectures. The PRODNET project aims to provide functionalities related to the creation and maintenance phases (search and selection of partners, negotiation, contracts awarding, tender preparation). The aim of the VEGA* project is to develop a software system that supports small and medium sized enterprises (SMEs) to set up and manage virtual enterprises as easily and quickly as possible.

Other projects address particular aspects in a specific phase of the VO operation process. For example, Multiagent Manufacturing Agile Scheduling Systems for Virtual Enterprises (MASSYVE) [18] focuses on agile scheduling, X-CITTIC (Planning and Control System for Semiconductor Virtual Enterprises) [22] concentrates on planning and controlling and STEP and the Virtual Enterprise (SAVE)[4] focuses on data modelling for VE. Furthermore, earlier research in enterprise-wide business management systems has focussed on the management (through automated negotiation) of business processes within a static organisational structure—ADEPT [11]—and models and techniques for information interchange—KRAFT [17].

6. Conclusions

A flexible mechanism for the formation of VOs has been presented in this paper that combines constraint solving, market clearing and quality modelling techniques. This model has a number of significant advantages. First, through quality modelling and the use of expressive representations of service packages, the CONOISE system may be deployed in realistic electronic commerce scenarios. Second, through the use of state-of-the art market clearing algorithms, VOs formed by CONOISE can be guaranteed to contain the optimal (or very close to the optimal) set of agents. Finally, taken in the context of the wider VO management process the VO formation mechanism presented in this paper represents a critical element of a flexible and robust solution to the problem of automating the management of virtual organisations.

Acknowledgements

The CONOISE project is funded by BT Exact - British Telecom's research, technology and IT operations business.

References

[1] P. Baptiste, C.L. Pape, W. Nuijten, Constraint-based scheduling: applying constraint programming to scheduling problems, In: International Series in Operations Research and Management Science, vol. 39, 2001.

[2] M. Burgess, W.A. Gray, N. Fiddian, Establishing a taxanomy of quality for use in information filtering, In: Proceedings of the 1996 Joint International Conference and Symposium on Logic Programming, 2002.

[3] L.M. Camarinha-Matos, H. Afsarmanesh, in: A. Molina, A. Kusiak, J. Sanchez (Eds.), Virtual enterprises: life cycle supporting tools and technologies, Handbook of Life Cycle Engineering: Concepts, Models and Technologies, Academic Publishers, New York, 1998, pp. 535–571.

[4] L.M. Camarinha-Matos, H. Afsarmanesh (Eds.), IFIP Conference Proceedings, vol. 153, Academic Publishers, New York, 1999.

[5] L.M. Camarinha-Matos, H. Afsarmanesh, R.J. Rabelo (Eds.), IFIP Conference Proceedings, vol. 184, Academic Publishers, New York, 2000.

[6] Y. Caseau, F. Laburthe, Cumulative scheduling with task intervals, In: Proceedings of the 1996 Joint International Conference and Symposium on Logic Programming, 1996, pp. 363–377.

[7] V.D. Dang, N.R. Jennings, Optimal clearing algorithms for multi-unit single item and multi-unit combinatorial auctions with demand/supply function bidding, In: Proceedings of the Fifth International Conference on Electronic Commerce, 2003, pp. 25–30.

[8] V.D. Dang, N.R. Jennings. Polynomial algorithms for clearing multi-unit single-item and multi-unit combinatorial auctions. Artificial Intelligence, 2004 (in press).

[9] M. Eso, S. Ghosh, J. Kalagnanam, L. Ladanyi. Bid evaluation in procurement auctions with piece-wise linear supply curves. Technical Report, IBM Research Report RC 22219, 2001.

[10] P. Faratin. Automated service negotiation between autonomous computational agents. PhD thesis, Queen Mary University of London, 2001.

[11] N.R. Jennings, P. Faratin, T.J. Norman, P. O'Brien, B. Odgers, Autonomous agents for business process management, International Journal of Applied Artificial Intelligence 14 (2) (2000) 145–189.

[12] M. Luck, P. McBurney, C. Preist, Agent technology: enabling next generation computing (a roadmap for agent based computing), AgentLink (2003).

[13] L. Mui, A. Halberstadt, M. Mohtashemi, Notions of reputation in multi-agent systems: a review, In: Proceedings of First International Joint Conference on Autonomous Agents and Multi-Agent Systems, 2002, pp. 280–287.

[14] T.D. Nguyen, N.R. Jennings, Concurrent bi-lateral negotiation in agent systems, In: Proceedings of the Fourth DEXA Workshop on E-Negotiations, 2003.

[15] T.D. Nguyen, N.R. Jennings, A heuristic model for concurrent bi-lateral negotiations in incomplete information settings, In: Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence, 2003, pp. 1467–1469.

[16] S.A. Petersen, M. Gruninger, An agent-based model to support the formation of virtual enterprises, In: International ICSC Symposium on Mobile Agents and Multi-agents in Virtual Organisations and E-Commerce, 2000.

[17] A.D. Preece, K. Hui, A. Gray, P. Marti, T. Bench-Capon, D. Jones, Z. Cui, The KRAFT architecture for knowledge fusion and transformation, Knowledge-Based Systems 13 (2-3) (2000) 113–120.

[18] R.J. Rabelo, L.M. Camarinha-Matos, H. Afsarmanesh, Multiagent perspectives to agile scheduling, In: Proceedings of IEEE/IFIP International Conference on Balanced Automation Systems, 1998.

[19] T. Sandholm, S. Suri, Market clearability, In: Proceedings of The Seventeenth International Joint Conference on Artificial Intelligence, 2001, pp. 1145–1151.

[20] P. Sieber, J. Griese (Eds.), Proceedings of the First International VoNet Workshop, 1998.

[21] P. Sieber, J. Griese (Eds.), Proceedings of the Second International VoNet Workshop, 1999.

[4] http://www.epmtech.jotne.com/newsletter/ew199/5.html.

[22] A.L. Soares, J.P. deSousa, A.L. Azevedo, J.A. Bastos. Using an informal ontology in the development of a planning and control system-the case of the virtual enterprise. Technical Report, INESC Porto., 1996.

[23] B. Suter, The VEGA* cooperation platform: providing real support for virtual enterprises, In: Sieber and Griese, 1999.

[24] P.R. Wurman, Dynamic pricing in the virtual marketplace, IEEE Internet Computing 5 (2) (2001) 36–42.

[25] B. Yu, M. Singh, An evidential model of distributed reputation management, In: Proceedings of First International Joint Conference on Autonomous Agents and Multi-Agent Systems, 2002, pp. 294–301.