Acquiring domain knowledge for negotiating agents: a case of study

Jose J. Castro-Schez[1], Nicholas R. Jennings, Xudong Luo,
Nigel R. Shadbolt*

*School of Electronics and Computer Science, University of Southampton, Highfield,
Southampton S017 1BJ, UK*

Abstract

In this paper, we employ the *fuzzy repertory table technique* to acquire the necessary *domain knowledge* for software agents to act as sellers and buyers using a bilateral, multi-issue negotiation model that can achieve optimal results in semi-competitive environments. In this context, the seller's domain knowledge that needs to be acquired is the rewards associated with the products and restrictions attached to their purchase. The buyer's domain knowledge that is acquired is their requirements and preferences on the desired products. The knowledge acquisition methods we develop involve constructing three fuzzy repertory tables and their associated distinctions matrixes. The first two are employed to acquire the seller agent's domain knowledge; and the third one is used, together with an inductive machine learning algorithm, to acquire the domain knowledge for the buyer agent.
© 2003 Published by Elsevier Ltd.

1. Introduction

Autonomous agents are being advocated, built and deployed in an increasing array of complex, distributed application domains (Jennings, 2001). One of the main reasons for this burgeoning interest is that the computational model of autonomous entities, interacting in flexible ways, is both a natural and a powerful way of

*Corresponding author. Tel.: +44-23-8059-7682; fax: +44-23-8059-3313.

E-mail addresses: josejesus.castro@uclm.es (J.J. Castro-Schez), nrj@ecs.soton.ac.uk (N.R. Jennings), xl@ecs.soton.ac.uk (X. Luo), nrs@ecs.soton.ac.uk (N.R. Shadbolt).

[1] On sabbatical leave from Department Informatica, Escuela Superior de Informatica, Universidad de Castilla-La Mancha, Paseo de la Universidad, 4, 13071 Ciudad Real, Spain.

analysing, designing and building such systems (Jennings, 2000). To this end, a key form of interaction in multi-agent systems is *automated negotiation*—the process by which a group of agents come to a mutually acceptable agreement on some matter (Jennings et al., 2001). Given its importance, a wide range of models for automated negotiation have been developed; these include models for auctions, direct one-to-one negotiations and argumentation-based encounters. To date, however, research in this field has been almost exclusively concerned with the development of efficient and effective algorithms that enable agents to be successful and obtain acceptable outcomes. While this is clearly important, it is only part of the picture. In most cases, agents negotiate on behalf of their owner (which may be an individual or an organization). However, for this to be effective, agents must be able to adequately represent their owners' interests, preferences, and prejudices in the given domain such that they can negotiate faithfully on their behalf. However, at this time, little thought has been given to the problems of exactly what knowledge an owner needs to impart to their agent to achieve high fidelity negotiation behaviour, and how such knowledge can be effectively acquired from the owner. These are clearly serious shortcomings of existing research that need to be addressed if negotiating agents are to be widely used.

Against this background, we have instigated research to start bridging the knowledge specification and acquisition gap that exists between the owners of negotiating agents and the negotiation algorithms that their agents use. Specifically, in this paper we demonstrate how this can be achieved for a particular negotiation model that we use the CommonKADS methodology (Schreiber et al., 2000) to develop (Luo et al., 2003b). This model is for bilateral multi-issue negotiations and it uses fuzzy constraint satisfaction techniques to achieve optimal results in semi-competitive environments. This particular model is chosen because bilateral encounters are generally among the most valuable classes of negotiations. Moreover, the model is representative of the class of heuristic methods that have been applied to automated negotiation problems. In addition, of course, it is one with which we are familiar. However, we could have chosen any other negotiation model that is available in the literature as our point of departure.

The techniques that we propose for acquiring this knowledge are those of the fuzzy repertory table (Castro-Schez et al., 2003) (the justification for this choice is given in Section 5) together with an inductive machine learning algorithm.[2] Having completed the knowledge and acquisition processes for our negotiation model, our goal will be to extend the approach to other types of negotiation model. In so doing, we want to determine whether there are knowledge requirements that are common to most or indeed all negotiation models and, if so, what they are. Moreover, we also aim to discover or develop the knowledge acquisition techniques that are most suitable for acquiring this knowledge.

This paper advances the state-of-the-art in a number of important ways. From the perspective of negotiating agents, it is the first paper to highlight the importance of tackling the knowledge acquisition problem. Moreover, it is one of the few examples

---

[2] In all of this work we assume users truthfully reveal information to their agents.

of knowledge acquisition methods being explicitly integrated into agents. It also demonstrates how to go about acquiring the knowledge that is required for one particular negotiation model. From the perspective of knowledge acquisition, it demonstrates how fuzzy repertory tables and the corresponding fuzzy induction can be used in agent-oriented task achieving architectures.

The remainder of this paper is structured in the following manner. Section 2 recalls the fuzzy repertory table that will be used to acquire the necessary domain knowledge from human users. Sections 3 and 4 show the domain knowledge acquisition process of the fuzzy repertory table in operation for our negotiation model and its application to an accommodation renting scenario where a buyer agent (acting on behalf of a student) negotiates with a seller agent (acting on behalf of a real estate agent) for a deal of renting accommodation. Section 5 justifies our choice of the fuzzy repertory table as an appropriate technique for acquiring the necessary domain knowledge from human sellers and buyers. Finally, Section 6 concludes and outlines the avenues of further research.

2. Fuzzy repertory tables

In this section, we describe the *fuzzy repertory table* (FRT) technique (Castro-Schez et al., 2003), its basic concepts and the process of an interaction with a user are presented in Sections 2.1 and 2.2. Section 2.3 presents a trapezoidal representation that extends the classic repertory grid in that a user can assign various types of values to the attributes of elements. Section 2.4 presents the method for calculating similarity between elements (a method that is different from that of the classic repertory grid). Finally, Section 2.5 provides an outline of our knowledge acquisition system based on fuzzy repertory tables.

2.1. Basic concepts

The repertory grid is a knowledge acquisition technique devised originally by the clinical psychologist *George Kelly* (1955). The central concepts in this technique are: (1) *elements*: any kind of objects (e.g., people, things, events, experiences, tasks or processes; in particular, the products that the seller wants to sell or the buyer wants to buy in our negotiation problem); and (2) *constructs*: attributes of elements that represent the dimensions of similarity and differences between elements, and are used to characterize those elements (e.g., the attributes of the products in our negotiation problem). The most basic form for a repertory grid is a rectangular matrix with elements as columns and constructs as rows. Each row–column intersection contains a rating showing a degree to which the human user applied a given construct to a particular element. This rating is taken from a rating scale.

2.2. Interaction with a user

The typical type of interaction of a repertory grid technique with a user can be briefly described as follows.

Step 1: Identify the set of elements:

    1. Human user submits a set of elements. At least three elements are required for using the triadic method.
    2. System computes combinations of triads (set of three elements) and selects a triad randomly.

Step 2: Repeat the following steps until all triads are considered or the user stops:
Step 2.1: Define a construct from the triad:

    1. System selects (randomly) a triad.
    2. Human user selects an attribute as a construct, which can make one element different from the remaining two.
    3. Human user defines a scale and a name for the construct.

Step 2.2: Perform a rating for all elements:

    1. Human user rates all elements on the construct's scale.

Step 3: Data processing:
    1. System collects all element ratings and then computes similarities, orderings, discrimination trees, etc. (Atkin, 1974; Slater, 1977; Rathod, 1981).
    2. System collects all constructs and then computes ties, entropy measures, decision trees, rules, etc. (Shaw, 1980; Gaines and Shaw, 1986, 1992).
    3. System computes entailments that are logical implications or statistical covariations (Ford et al., 1991).

    Data are often presented as a datagram.
    Similarity-breaking:
Step 4: Construct-breaking—after analysis there may be elements that have very similar ratings:

    1. System shows the element similarity to the user.
    2. Allow user to do a similarity-break. User tries to identify a new construct that puts the similar elements on opposite poles of a new construct.
    3. Go to Step 2.2.

Step 5: Element-breaking—there are constructs with very similar ratings for elements:

    1. System shows the construct tie to the user.
    2. User tries to find (and add) a new element which breaks the similarity.
    3. Rate this new element on all other constructs.
    4. Go to Step 3.

Step 6: End.

2.3. Rating attributes using trapezoid numbers

A fuzzy repertory table (FRT) also looks like a rectangular matrix with elements (as columns) and constructs (as rows). Each row–column intersection contains a rating. Such a rating is a trapezoid number showing how a user applied a given construct to a particular element. A trapezoid number (a, b, c, d) is a fuzzy set that has a membership function of the following form:

$$\mu(x) = \begin{cases} 0 & \text{if } x < a, \\ \frac{x-a}{b-a} & \text{if } a \leqslant x < b, \\ 1 & \text{if } b \leqslant x \leqslant c, \\ \frac{d-x}{d-c} & \text{if } c < x \leqslant d, \\ 0 & \text{if } x > d. \end{cases} \tag{1}$$

By using trapezoid numbers, the FRT technique relaxes the restriction, of the classical repertory grid, that the ratings must be crisp numbers in a predefined range. In fact, trapezoid numbers enable the FRTs to provide categorical and numerical data types that may be given by means of linguistic terms (as shown in Fig. 1).

For instance, in the FRT developed in our accommodation renting scenario shown in Table 1, taking accommodation A_1 as an example, its attribute *district-type* is rated on a 0–4 rating scale (this rating provides an indication of user's preferences: 0—Does not like it and 4—Likes it). This room has no *new-furniture* nor an *air-conditioner* (both boolean attributes) but has a *phone* (boolean attribute), its *rental-rate* is `cheap` (continuous attribute), and it is located in the St. Dennis district (similar rating scale to the lodging type attribute). Each value is expressed by a membership function that is determined from the construct type (see Fig. 1) and direct interaction with the user.
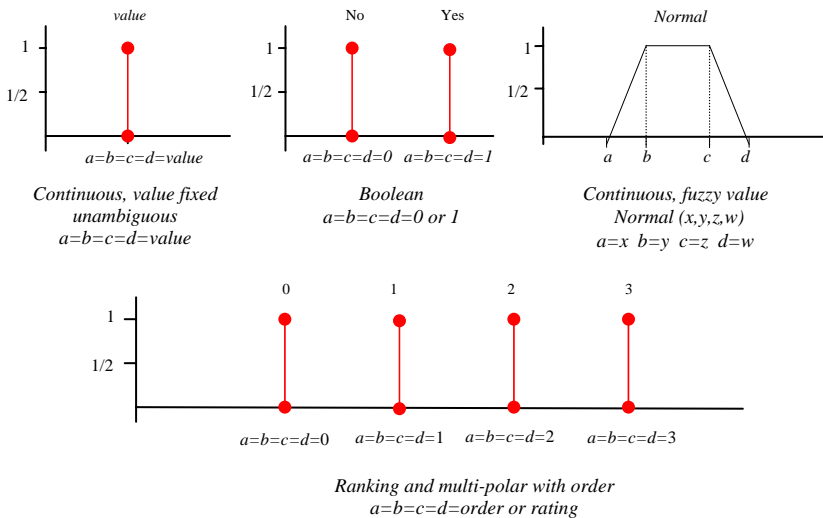

Fig. 1. Various types of rates that trapezoid numbers can represent.

Table 1
FRT developed in our accommodation renting scenario for acquiring rewards associated with each accommodation

|	A_1	A_2	A_3	A_4	A_5	A_6	A_7
Lodging type	Room in guest house $a = b = c = d = 2$	Room in guest house $a = b = c = d = 2$	Flat $a = b = c = d = 3$	Shared house $a = b = c = d = 1$	Apartment $a = b = c = d = 4$	Shared room $a = b = c = d = 0$	Shared house $a = b = c = d = 1$
New Furniture	No $a = b = c = d = 0$	No $a = b = c = d = 0$	No $a = b = c = d = 0$	No $a = b = c = d = 0$	No $a = b = c = d = 0$	Yes $a = b = c = d = 1$	No $a = b = c = d = 0$
Air conditioning	No $a = b = c = d = 0$	No $a = b = c = d = 0$	Yes $a = b = c = d = 1$	No $a = b = c = d = 0$	Yes $a = b = c = d = 1$	No $a = b = c = d = 0$	No $a = b = c = d = 0$
Rental-rate	Cheap $a = 300$ $b = 350$ $c = 400$ $d = 450$	Cheap $a = 300$ $b = 350$ $c = 400$ $d = 450$	Expensive $a = 550$ $b = 600$ $c = 650$ $d = 700$	Normal $a = 400$ $b = 450$ $c = 550$ $d = 600$	Very expensive $a = 650$ $b = 700$ $c = 1000$ $d = 1000$	Very cheap $a = 100$ $b = 150$ $c = 300$ $d = 350$	Normal $a = 400$ $b = 450$ $c = 550$ $d = 600$
Phone	Yes $a = b = c = d = 1$	Yes $a = b = c = d = 1$	No $a = b = c = d = 0$	No $a = b = c = d = 0$	Yes $a = b = c = d = 1$	No $a = b = c = d = 0$	No $a = b = c = d = 0$
District	St. Denis $a = b = c = d = 1$	Basset $a = b = c = d = 4$	Highfield $a = b = c = d = 2$	St. Denis $a = b = c = d = 1$	Basset $a = b = c = d = 4$	Bitterne $a = b = c = d = 0$	Basset $a = b = c = d = 4$
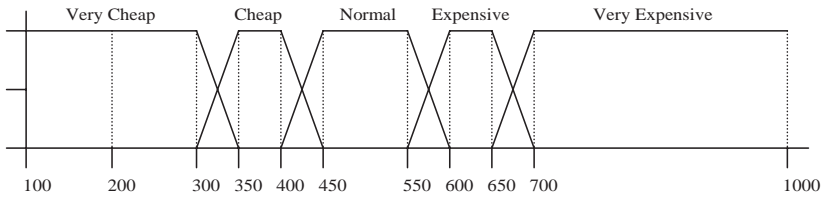
Fig. 2. Definition of the attribute *rental-rate* (£).

In the example shown in Table 1, during the course of developing the FRT, the attribute *rental-rate* takes on linguistic terms since the user found it more natural to describe it with a vague value. In particular, the set of linguistic terms that the attribute *rental-rate* takes on is shown in Fig. 2.

2.4. Assessing similarity

In order to assist in the further elicitation of attributes (or elements), or to refine the values already given to defined attributes, it is necessary to assess the pairwise similarity of known elements or constructions. The manner in which the FRT assesses similarity among elements is different from that of the classic repertory grid although their interaction with the user is similar. In fact, the FRT technique uses a separation measure for assessing the similarity between elements. The separation between two elements e_x and e_y against one construct/attribute a_i is given by the area between value A of attribute a_i of e_x and value B of attribute a_i of e_y. Formally, we have:

Definition 1. Let trapezoid number $\tau_{x(a_i)} = (a, b, c, d)$ be the rate for attribute a_i of element e_x and trapezoid number $\tau_{y(a_i)} = (a', b', c', d')$ be the rate for attribute a_i of element e_y. Then the measure of separation between e_x and e_y with respect to $\tau_{x(a_i)}, \tau_{y(a_i)}$ and a_i is:

$$d(e_x, e_y, \tau_{x(a_i)}, \tau_{y(a_i)}, a_i) = p_1 + p_2, \tag{2}$$

where

$$p_1 = \begin{cases} \dfrac{b' - c + a' - d}{2} & \text{if } d \leqslant a', \\ \dfrac{(b' - c) \times h}{2} & \text{if } a' < d \text{ and } c \leqslant b', \\ 0 & \text{otherwise}; \end{cases}$$

$$p_2 = \begin{cases} \dfrac{b - c' + a - d'}{2} & \text{if } d' \leqslant a, \\ \dfrac{(b - c') \times h'}{2} & \text{if } a < d' \text{ and } c' \leqslant b, \\ 0 & \text{otherwise}; \end{cases}$$

$$h = \frac{b' - c}{b' - a' + d - c};$$

$$h' = \frac{b - c'}{b - a + d' - c'}.$$

In the example shown in Tables 1 and 3, accommodation A_2 takes the linguistic term `cheap`= (300,350,400,450), and A_5 takes the linguistic term `very-expensive`= (650,700,1000,1000). Thus, by (2) the separation between these elements with respect to attribute *rental-rate* and its two rates `cheap` and `very-expensive` (see Fig. 3) is:

$$d(A_2, A_5, \texttt{cheap}, \texttt{very-expensive}, \textit{rental-rate})$$

$$= p_1 + p_2$$
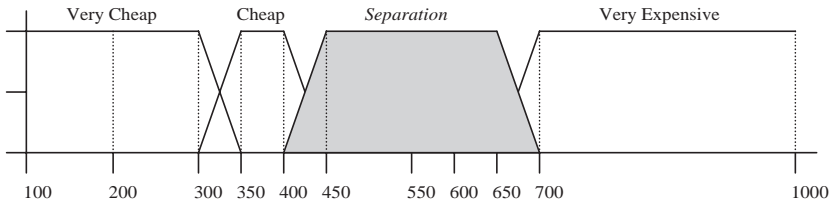$$= \frac{700 - 400 + 650 - 450}{2} + 0$$
$$= 250.$$

Fig. 3. Example of separation measurement between examples of accommodation A_2 and A_5 from Table 1, according to the construct (attribute) *rental-rate*.

Definition 2. Let trapezoid number $\tau_{x(a_i)} = (a, b, c, d)$ be the rate for attribute a_i of element e_x and trapezoid number $\tau_{y(a_i)} = (a', b', c', d')$ be the rate for attribute a_i of element e_y. Then the normalized measure of separation between e_x and e_y with respect to $\tau_{x(a_i)}, \tau_{y(a_i)}$ and a_i is:

$$d_N(e_x, e_y, \tau_{x(a_i)}, \tau_{y(a_i)}, a_i) = \frac{d(e_x, e_y, \tau_{x(a_i)}, \tau_{y(a_i)}, a_i)}{d(e_x, e_y, \tau_{max}(a_i), \tau_{min}(a_i), a_i)}, \tag{3}$$

where $d(e_x, e_y, \tau_{x(a_i)}, \tau_{y(a_i)}, a_i)$ is given by (2), $\tau_{max}(a_i)$ is a trapezoid value whose first two parameters are not smaller than those of other possible trapezoid values that are used to rate attribute a_i, and $\tau_{min}(a_i)$ is a trapezoid value whose first two parameters are not greater than those of other possible trapezoid values that are used to rate attribute a_i.

For example, in Table 1 the normalized separation between A_2 and A_5 with respect to attribute *rental-rate* and its two rates `cheap` and `very-expensive` is:

$$d_N(A_2, A_5, \texttt{cheap}, \texttt{very-expensive}, \textit{rental-rate})$$

$$= \frac{d(A_2, A_5, \text{cheap}, \text{very-expensive}, \textit{rental-rate})}{d(A_2, A_5, \text{very-cheap}, \text{very-expensive}, \textit{rental-rate})}$$

$$= \frac{250}{\frac{700 - 300 + 650 - 350}{2}}$$

$$= 0.71.$$

The overall separation between two elements is the sum of the normalized separation between those two elements against each attribute that defines them. Formally, we have:

Definition 3. The overall separation between two elements e_x and e_y is:

$$D(e_x, e_y) = \frac{\sum_{i=1}^{m} d_N(e_x, e_y, \tau_{x(a_i)}, \tau_{y(a_i)}, a_i)}{m}, \tag{4}$$

where m is the number of attributes and $d_N(e_x, e_y, \tau_{x(a_i)}, \tau_{y(a_i)}, a_i)$ is given by (3).

For example, in Table 1, the overall separation between A_2 and A_5 is:

$$D(A_2, A_5) = \frac{\sum_{i=1}^{6} d_N(A_2, A_5, \tau_{A_2(a_i)}, \tau_{A_5(a_i)}, a_i)}{6}$$

$$= \frac{\frac{2}{4} + 0 + 1 + \frac{250}{350} + 0 + 0}{6}$$

$$= 0.37.$$

In the process of developing a FRT, the *distinctions matrix* plays an important role.

Definition 4. The distinctions matrix is a bi-dimensional matrix with $n \times n$ cells (see Table 2), where n is the number of FRT elements. Each cell (i, j) contains the attributes that can distinguish between elements e_i and e_j, and the strength to which they are distinguished. The strength to which a construct distinguishes between elements e_i and e_j is the normalized separation between those two elements against that attribute.

For example, in the distinctions matrix shown in Table 2, the cell (2,3) contains the following information: the attributes that distinguish A_2 from A_3 are C_3 (*air-conditioner*) and C_5 (*phone*), and the strength of such distinctions are 1 (this is the maximum strength in normalized space) because A_3 has *air-conditioner* and A_2 does not and A_2 accommodation has a *phone* whereas A_3 does not.

2.5. The overview of our knowledge acquisition method

In the following two sections, we will detail how the FRT and the distinctions matrix are employed to acquire the seller's and the buyer's domain knowledge. The proposed method includes three FRTs: FRT1, FRT2 and FRT3 (see Fig. 4). FRT1

Table 2
Distinctions matrix of the FRT shown in Table 1, with C_i (α) being the attribute i, its meaning in the cell (x,y) is that i separates elements x and y to a degree α (see(3))

|	A_1	A_2	A_3	A_4	A_5	A_6	A_7
A_1	Nil	$C_6(0.75)$;	$C_3(1);C_5(1)$;	$C_5(1)$;	$C_3(1);C_4(0.71)$; $C_6(0.75)$;	$C_2(1);C_5(1)$;	$C_5(1)$; $C_6(0.75)$;
A_2	$C_6(0.75)$;	Nil	$C_3(1);C_5(1)$;	$C_5(1)$; $C_6(0.75)$;	$C_3(1);C_4(0.71)$;	$C_2(1);C_5(1)$; $C_6(1)$;	$C_5(1)$;
A_3	$C_3(1);C_5(1)$;	$C_3(1);C_5(1)$;	Nil	$C_3(1)$;	$C_5(1)$;	$C_1(0.75);C_2(1)$; $C_3(1);C_4(0.71)$;	$C_3(1)$;
A_4	$C_5(1)$;	$C_5(1)$; $C_6(0.75)$;	$C_3(1)$;	Nil	$C_1(0.75);C_3(1)$; $C_5(1);C_6(0.75)$;	$C_2(1)$;	$C_6(0.75)$;
A_5	$C_3(1);C_4(0.71)$; $C_6(0.75)$;	$C_3(1)$; $C_4(0.71)$;	$C_5(1)$;	$C_1(0.75)$; $C_3(1);C_5(1)$; $C_6(0.75)$;	Nil	$C_1(1);C_2(1)$; $C_3(1);C_4(1)$; $C_5(1);C_6(1)$;	$C_1(0.75)$; $C_3(1);C_5(1)$;
A_6	$C_2(1);C_5(1)$;	$C_2(1);C_5(1)$; $C_6(1)$;	$C_1(0.75);C_2(1)$; $C_3(1);C_4(0.71)$;	$C_2(1)$;	$C_1(1);C_2(1)$; $C_3(1);C_4(1)$; $C_5(1);C_6(1)$;	Nil	$C_2(1)$; $C_6(1)$;
A_7	$C_5(1);C_6(0.75)$;	$C_5(1)$;	$C_3(1)$;	$C_6(0.75)$;	$C_1(0.75);C_3(1)$; $C_5(1)$;	$C_2(1);C_6(1)$;	Nil
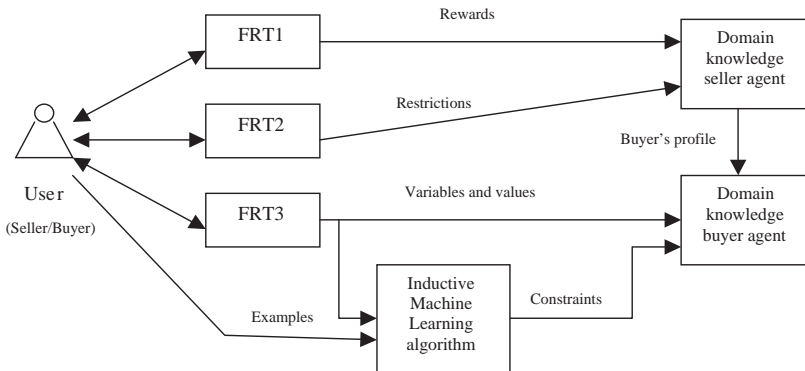
Fig. 4. Schema of our knowledge acquisition method for negotiating agents.

and FRT2 are used to acquire, from human sellers, the seller agent's domain knowledge—rewards and restrictions associated with each product, respectively. FRT3 is used with an inductive machine learning algorithm to acquire the buyer's domain knowledge of preferences and requirements upon desired products.

3. Seller's domain knowledge acquisition

In this section we first recall the notation of the seller's domain knowledge in our negotiation model (Luo et al., 2003b), and then discuss how to use the FRT method to acquire the rewards associated with products and the restrictions attached to products in the seller's domain knowledge.

3.1. Domain knowledge of the seller agent

The seller agent consists of five main components: (1) the domain knowledge—a set of products plus their associated information, (2) the inference knowledge—a set of primitive actions, (3) the task knowledge—a control regime, (4) the communication port—an interaction protocol, and (5) the working memory. Since this paper concentrates on discussing the acquisition of seller's domain knowledge, we only recall the definition of the domain knowledge and omit the other knowledge components (however, the full details can be found in Luo et al., 2003b).

Definition 5. The domain knowledge of the seller agent is the set of *products* it holds:

$$\mathcal{G} = \{g_i | g_i = (c_i, r_i, u_i, p_i), p_i = (v_{i1}, \ldots, v_{in}), 0 \leqslant i \leqslant k\},$$

where the following hold:

- c_i is the *restriction* attached to product g_i that a buyer agent must satisfy in order to obtain the product (e.g., buyers must be over 18 years old).
- r_i is the *reward* associated with product g_i, which the seller agent may use to persuade a buyer agent to purchase the product.
- u_i is the *profit* that the seller agent gets if product g_i is sold at a particular price. This is private information that the seller will not reveal to the buyer.
- p_i, called the *product-attributes*, is the value vector of negotiable attributes (e.g., price, quality, model, volume, delivery date, expiry date, after-sale service and warranty) of product g_i.
- k is the total number of products the seller agent possesses.

In the seller's domain knowledge, the concepts of *reward* and *restriction* are essential. The rewards are features that distinguish between the seller's products and other competitors' products or standard products; and the restrictions attached to each product are distinctive features that potential buyers must satisfy in order to obtain this product. This paper concentrates on acquiring these two kinds of seller's domain knowledge. In fact, we shall use the FRT and the information in the corresponding distinctions matrix to detect attributes that can be used, in the negotiation process, as the rewards and restrictions of a product.

3.2. Acquiring rewards associated with each product

According to the repertory grid method given in Section 2.1, we shall describe the procedure of acquiring the rewards associated with each product as follows. Firstly, the elements of FRT1 are the products that a seller can offer. We denote the set of products as $\mathscr{E}$. In the accommodation renting scenario, the elements of FRT1 are available properties of the real estate agent. In the example of Table 1,

$$\mathscr{E} = \{A_1, ..., A_7\}. \tag{5}$$

We assume that similar products can be offered by other sellers (the seller's competitors).

Secondly, in constructing FRT1 the human seller is asked to describe and analyse the differences and similarities among products in $\mathscr{E}$. To do this, FRT1 uses the following question.

> *How are two products similar and different from the third in the way in which they are offered to the buyer?*

The human seller answers this question by using the attributes of the products. Thus, through interaction with the human seller, FRT1 obtains a set of attributes that identify each one of the products in $\mathscr{E}$. In the accommodation renting scenario, the acquired knowledge from FRT1 is shown in Table 1.

Thirdly, we compute the similarity among the acquired elements and constructs of FRT1 in order to identify the reward associated with each product. Table 2 shows the distinctions matrix associated with the acquired knowledge via FRT1 in our scenario.

Fourthly, the system identifies the reward (r_i) associated with each product. This can be done via the distinctions matrix since this matrix holds the distinctions among elements in $\mathscr{E}$. In fact, attributes that can distinguish a product offered by the seller from other sellers' products, could be used by the seller as advantageous features for selling his products. From Table 2, we can see that in our scenario, accommodation A_6 is distinguished from the others in terms of attribute C_2 (*new-furniture*) since A_6 is the only accommodation that the seller offers with new furniture (see Table 1). In the same scenario, although *phone* (C_5) is an attribute that can distinguish A_3 from most other accommodation, it cannot be considered a reward associated with accommodation A_3 since it is negative for A_3, i.e., it has no phone (see Table 1) (although of course some might regard that as a positive feature).

Notice that at the current stage, our fuzzy repertory table cannot automatically distinguish between positive and negative attributes. Thus, the attributes obtained through FRT1 are just reward candidates associated with each product. As a result, the human seller must review these reward candidates and then manually select and promote appropriate attributes to be rewards.

From FRT1 (Table 1) and its associated distinctions matrix (Table 2), we can obtain the following information (see Castro-Schez et al., 2003).

- Accommodation A_1 has no *air-conditioner* but has a *phone*, and is located in the St. Dennis district.
- Accommodation A_2 has no *air-conditioner* but has a *phone*, and is located in the Bassett district.
- Accommodation A_3 has an *air-conditioner* but no *phone*.
- Accommodation A_4 has no *new-furniture*, no *air-conditioner*, and *no phone*, and is located in the St. Dennis district.
- Accommodation A_5 has an *air-conditioner* and a *phone*.
- Accommodation A_6 has *new-furniture*.
- Accommodation A_7 has no *air-conditioner* and no *phone*, and is located in the Bassett district.

Finally, from the above information the human seller selects those attributes that distinguish one product from the others in a positive sense as the rewards associated with each product. For our renting accommodation scenario, Table 3 shows the knowledge about rewards associated with each product, which we acquire in this way. From Table 3, we notice that the seller considered *located in Bassett district* as a reward associated to A_7. This is because it is considered a desirable place to live.

3.3. Acquiring restrictions attached to each product

FRT2 is used to acquire restrictions, c_i, attached to each product g_i offered by the seller. The set of elements of FRT2 is the same as that of FRT1.

Firstly, a human seller has to complete a FRT session to acquire the differences and similarities among the product set considering who is eligible to buy each product. To do so, the human seller needs to answer the following question.

How are two products similar and different from the third in terms of who is eligible to purchase them ?

Thus, for each product, the set of features that potential buyers would demand is acquired.

Next, restrictions c_i attached to each product g_i can be identified through the distinctions matrix associated with FRT2. Table 4 shows a FRT2 example, and

Table 3
Rewards associated with each accomodation in examples of Table 1

Accommodation	Associated rewards
A_1	Phone
A_2	Phone $\bigwedge$ placed in Bassett district
A_3	Air-conditioner
A_4	No reward
A_5	Phone $\bigwedge$ air-conditioner
A_6	New-furniture
A_7	Placed in Bassett district

Table 4
FRT developed for acquiring restrictions of our buyers

	A_1	A_2	A_3	A_4	A_5	A_6	A_7
Foreigner	Yes $a = b = c = d = 1$	No $a = b = c = d = 0$	Yes $a = b = c = d = 1$	Yes $a = b = c = d = 1$	No $a = b = c = d = 0$	Yes $a = b = c = d = 1$	Yes $a = b = c = d = 1$
Sex	Female $a = b = c = d = 0$	Any	Any	Male $a = b = c = d = 1$	Any	Male $a = b = c = d = 1$	Female $a = b = c = d = 0$
Does buyer like cook?	Any	Any	No $a = b = c = d = 0$	Yes $a = b = c = d = 1$	Any	Yes $a = b = c = d = 1$	Any
Rental period	$x < = 6$ $a = 1$ $b = 1$ $c = 6$ $d = 8$	Any	$x >= 24$ $a = 24$ $b = 26$ $c = 36$ $d = 36$	$24 > x >= 12$ $a = 12$ $b = 14$ $c = 24$ $d = 26$	Any	Any	$12 > x >= 6$ $a = 6$ $b = 8$ $c = 12$ $d = 14$
Does buyer like pet?	No $a = b = c = d = 0$	No $a = b = c = d = 0$	Any	No $a = b = c = d = 0$	Any	No $a = b = c = d = 0$	Yes $a = b = c = d = 1$
Occupation	Any	Student	A professional	Any	A professional	Student	A professional

Table 5
Distinctions matrix associated with FRT from Table 4

	A_1	A_2	A_3	A_4	A_5	A_6	A_7
A_1	Nil	$C_1(1)$;	$C_4(1)$;	$C_2(1)$;	$C_1(1)$;	$C_2(1)$;	$C_5(1)$;
A_2	$C_1(1)$;	Nil	$C_1(1)$; $C_6(1)$;	$C_1(1)$;	$C_6(1)$;	$C_1(1)$;	$C_1(1)$; $C_6(1)$; $C_5(1)$;
A_3	$C_4(1)$;	$C_1(1)$; $C_6(1)$;	Nil	$C_3(1)$;	$C_1(1)$;	$C_3(1)$; $C_6(1)$;	$C_4(0.67)$;
A_4	$C_2(1)$;	$C_1(1)$;	$C_3(1)$;	Nil	$C_1(1)$;		$C_2(1)$; $C_5(1)$;
A_5	$C_1(1)$;	$C_6(1)$;	$C_1(1)$;	$C_1(1)$;	Nil	$C_1(1)$; $C_6(1)$;	$C_1(1)$;
A_6	$C_2(1)$;	$C_1(1)$;	$C_3(1)$; $C_6(1)$;		$C_1(1)$; $C_6(1)$;	Nil	$C_2(1)$; $C_6(1)$; $C_5(1)$;
A_7	$C_5(1)$;	$C_1(1)$; $C_6(1)$; $C_5(1)$;	$C_4(0.67)$;	$C_2(1)$; $C_5(1)$;	$C_1(1)$;	$C_2(1)$; $C_6(1)$; $C_5(1)$;	Nil

Table 5 shows its associated distinctions matrix. From Tables 4 and 5, we can obtain the following information (see Castro-Schez et al., 2003).

- Accommodation A_1 can be rented by a tenant of any nationality but who should be female and is not interested in keeping pets, and whose rental period should be less than 6 months.
- Accommodation A_2 can be rented only by native tenants who are students and not interested in keeping pets.
- Accommodation A_3 can be rented by a tenant of any nationality but who should be a professional and does not need to cook, and whose rental period should be longer than 24 months.
- Accommodation A_4 can be rented by a tenant of any nationality but who should be male and needs to cook.
- Accommodation A_5 can be rented only by a native tenant who should be a professional.
- Accommodation A_6 can be rented by a tenant of any nationality but who should be a male student.
- Accommodation A_7 can be rented by a tenant of any nationality but who should be a female professional and would like to keep a pet, and whose rental period is approximately between 6 and 12 months.

The above information obtained from FRT2 is shown to the human seller to identify restrictions attached to each product according to the following principles.

The restrictions attached to each product are the attributes that can distinguish the product from the rest of products (appear in the distinctions matrix) and that take a

Table 6
Restrictions attached to each accommodation after seller's analysis

Accommodation	Attached restrictions
A_1	Female $\bigwedge$ rental-period $\leqslant 6$ months $\bigwedge$ no pets allowed
A_2	NOT foreigner $\bigwedge$ students $\bigwedge$ no pets allowed
A_3	Cooking not allowed $\bigwedge$ rental-period $\geqslant 24$ months $\bigwedge$ professional
A_4	Male $\bigwedge$ no pets allowed
A_5	NOT foreigner $\bigwedge$ professional
A_6	Male $\bigwedge$ student $\bigwedge$ no pets allowed
A_7	Female $\bigwedge$ 6 months $\leqslant$ rental-period $\leqslant 12$ months $\bigwedge$ professional

value that is significant for the seller in terms of restricting the buyer who can purchase the product.

Thus, in the accommodation renting scenario, although *foreigner* is an attribute that can distinguish A_4 from A_2 and A_5 (see Table 5), this attribute cannot be regarded as a restriction attached to accommodation A_4 because it does not take a suggestive value (it can be rented by a tenant of any nationality). However, we can say that *foreigner* can be regarded as a restriction attached to accommodation A_2 and A_5 because it takes a suggestive value (it can be only rented by a native tenant). Table 6 shows the acquired information from the accommodation renting scenario.

4. Buyer's domain knowledge acquisition

In this section, we firstly recall the notation of the buyer's domain knowledge, and then present our method for acquiring it from a human buyer.

4.1. Domain knowledge of the buyer agent

Our buyer agent consists of five similar components to the seller: (1) a domain knowledge model consisting of the buyer's requirement/preference model and the buyer's profile model,[3] (2) an inference model corresponding to a set of primitive actions, (3) a task knowledge model that specifies the control regime, (4) a communication port, and (5) a working memory. Since this paper focuses on domain knowledge acquisition, in the following we only recall the component of buyer's domain knowledge (however, as before, the details of other components can be found in Luo et al., 2003b).

We use prioritized fuzzy constraints to express buyers' requirements and preferences concerning the desired products. The concept of prioritized fuzzy constraint problems (PFCSPs) was introduced by Dubois et al., (1994, 1996); Dubois

---
[3] We assume both of these are fixed for the duration of the negotiation encounter.

and Prade, (1999); and the following is a revised version of their formulation (see Luo et al. (2003a) for the motivation and technical details of the extension).

Definition 6. A prioritized fuzzy constraint satisfaction problem (PFCSP) is defined as a 4-tuple (X, D, C^f, ρ), where the following hold.

(1) $X = \{x_i | i = 1, \ldots, n\}$ is a finite set of *variables*.
(2) $D = \{d_i | d_i$ is the domain on which the variable x_i takes values, $i = 1, \ldots, n\}$ is a finite set of all *domains* associated with each variable in X.
(3) C^f is a set of *fuzzy constraints*:

$$C^f = \left\{ R_i^f | \mu_{R_i^f} : \left(\prod_{x_j \in var(R_i^f)} d_j \right) \to [0, 1], i = 1, \ldots, m \right\}, \tag{6}$$

where $var(R_i^f)$ denotes the set of variables of R_i^f.
(4) $\rho : C^f \to [0, \infty)$ is a *priority function*. This function puts the constraints into order of importance so that the most important one has the highest value.

Definition 7. The domain knowledge of the buyer agent is $\mathcal{KD} = (\mathcal{C}, \mathcal{B})$ where:

- $\mathcal{C} = (X, D, C^f, \rho)$ is the *buyer's requirement/preference model*. (X, D, C^f, ρ) is a PFCSP (see Definition 6). X is the set of attributes of the products. Each domain $d_i \in D$ is a set of possible values of an attribute of the products. C^f is a set of fuzzy constraints that express the buyer's requirements and preferences on the attributes of the desired product. Each constraint $R_i^f \in C^f$ is associated with a *priority* $\rho(R_i^f) \in [0, +\infty)$.
- $\mathcal{B} = (F, t)$ is a fuzzy truth proposition system, called the *buyer's profile model*, which describes the background information it uses to evaluate the seller's offer. $F = \{f_i | i = 1, \ldots, l\}$ is a set of fuzzy propositions. $t : F \to [0, 1]$ is a truth function.

4.2. Buyer's requirement and preference acquisition

We employ a new FRT, called FRT3, and an inductive machine learning algorithm (Castro et al., 1999) for acquiring fuzzy constraints that represent the buyer's requirements and preferences. More concretely, firstly we use FRT3 to acquire the attributes and their possible values that are used by the buyer to express their requirements and preferences. Then, we apply the machine learning algorithm mentioned above, using the attributes and values acquired via FTR3, to acquire the prioritized fuzzy constraints.

The process of FRT3 is as follows. Firstly, the elements of this FRT ($\mathcal{E}$) are a set of standard products that could be offered by a seller or proposed by a buyer. In our example, the set consists of accommodation taken from a database whose elements are reasonably representative of the market:

$$\mathcal{E} = \{a_1, \ldots a_n\}.$$

Next, the human buyer is asked to describe and analyse the differences and similarities among products from the set $\mathscr{E}$. This is achieved by asking the human buyer the following question.

> *How are two products similar and different from the third in a way that would matter to you when you* consider renting them?

The human buyer answers this question by means of those attributes that he would use when deciding upon an offered product. Unlike FRT1 and FRT2, the element set of FRT3 is not fixed. It will vary as the FRT3 elicitation progresses. Thus, in order to acquire an attribute that can distinguish some products from other ones, the human buyer is asked the following.

> *What is a good qualification from this attribute? Or what would be good values for this attribute? Or what value of this attribute would be ideal?*

From the set $\mathscr{E}$ we can extract those products that are distinguished strongly from others in terms of this attribute value. Then, the human buyer is asked to estimate the importance or priority of this attribute when deciding whether to purchase a product. In the early phases of FRT3 elicitation, higher priorities are recommended since human buyers often like to give the most important attributes first. This attribute can be introduced in the set X and values employed for rating this attribute in the set D. FRT3 is continuously developed with new examples. This process is continued until we obtain a set of examples that a buyer would be disposed to purchase or the buyer stops the process.

In our example of accommodation renting, the initial set $\mathscr{E}$ is presented to a human buyer. The obtained FRT3 is shown in Table 7. Then from Table 7, we obtain the information: *rental-rate* is an attribute that can distinguish in a strong and significant way between accommodation in the element set $\mathscr{E}$. Further, the human buyer says that he would rent cheap accommodation rather than expensive ones. Thus, this attribute is introduced in the set X. The attribute *rental-rate* is fuzzy continuous and its possible values are linguistic terms:

cheap $= (100, 100, 250, 300)$, normal $= (250, 300, 400, 450)$,

expensive $= (400, 450, 600, 600)$

that constitute its domain set D. Each linguistic term, as a fuzzy set, is associated with a membership function whose parameters are given between parentheses. The

Table 7
FRT3 developed for acquiring the buyer's requirements, phase 1

	a_1	a_2	a_3	a_4	a_5	...	a_n
	Cheap	Cheap	Cheap	Normal	Expensive	...	Expensive
Rental rate	$a = 100$	$a = 100$	$a = 100$	$a = 250$	$a = 400$		$a = 400$
	$b = 100$	$b = 100$	$b = 100$	$b = 300$	$b = 450$		$b = 450$
	$c = 250$	$c = 250$	$c = 250$	$c = 400$	$c = 600$		$c = 600$
	$d = 300$	$d = 300$	$d = 300$	$d = 450$	$d = 600$		$d = 600$

Table 8
FRT3 developed for acquiring the buyer's requirements, phase 2

	a_1	a_2	a_3	a_4	a_5	…	a_m
Rental rate	Cheap $a = 100$ $b = 100$ $c = 250$ $d = 300$	Cheap $a = 100$ $b = 100$ $c = 250$ $d = 300$	Cheap $a = 100$ $b = 100$ $c = 250$ $d = 300$	Normal $a = 250$ $b = 300$ $c = 400$ $d = 450$	Normal $a = 250$ $b = 300$ $c = 400$ $d = 450$	…	Cheap $a = 100$ $b = 100$ $c = 250$ $d = 300$
Distance to work place	Far $a = 30$ $b = 35$ $c = 50$ $d = 50$	Near $a = 0$ $b = 0$ $c = 15$ $d = 25$	Average $a = 15$ $b = 25$ $c = 30$ $d = 35$	Far $a = 30$ $b = 35$ $c = 50$ $d = 50$	Near $a = 0$ $b = 0$ $c = 15$ $d = 25$	…	Near $a = 0$ $b = 0$ $c = 15$ $d = 25$

human buyer is asked to assign a priority to this attribute, and thus, for example, the buyer assigns 20 to the attribute priority.

Once an important attribute (i.e., *rental rate*) and its most desirable value (i.e., cheap) are detected, from $\mathscr{E}$ we can extract the products that do not meet this criterion (Castro-Schez et al., 2003). The set of extracted products is denoted as $\mathscr{E}'$. We continue to develop FRT3 with the new set of products, that is $\mathscr{E} - \mathscr{E}'$:

$$\mathscr{E} = \{a_1, ..., a_{n-|\mathscr{E}'|}\}.$$

The obtained FRT3 is shown in Table 8. From this table, it can be seen that *distance-to-working-place* is the next attribute that can distinguish among products in $\mathscr{E}$. This attribute takes the following linguistic terms:

near $= (0, 0, 15, 25)$, average $= (15, 25, 30, 35)$, far $= (30, 35, 50, 50)$.

The most desirable value of this attribute for the human buyer is *near* meaning that the desired accommodation is located *near* to the working place. Attribute *distance-to-working-place* is added into X and its possible values are added into the attribute's domain set D. Then, the human buyer is asked to assign a priority to this attribute, and thus, for example, he assigns the value 10. Finally, we extract from $\mathscr{E}$ the accommodations that are located *far* from the working place.

This process is repeated until the human buyer stops developing FRT3 or the human buyer is well disposed to purchase any product in FRT3. Finally, we obtain the FRT3 as shown in Table 9. The sets X and D that are acquired by using FRT3 are shown in Table 10.

Next, the buyer is shown a set of products assessed according to the attributes acquired by FRT3. The aim is to elicit a decision as to whether he would rent the accommodation. In this way, we obtain a set of examples that can be used by the inductive machine learning method proposed in (Castro et al., 1999) to acquire prioritised fuzzy constraints. In this case, the examples are represented as

Table 9
FRT developed for acquiring the buyer's requirements, final phase

	a_1	a_2	a_3	a_4	a_5	...	a_g
Rental rate	Cheap $a = 100$ $b = 100$ $c = 250$ $d = 300$	Cheap $a = 100$ $b = 100$ $c = 250$ $d = 300$	Cheap $a = 100$ $b = 100$ $c = 250$ $d = 300$	Normal $a = 250$ $b = 300$ $c = 400$ $d = 450$	Normal $a = 250$ $b = 300$ $c = 400$ $d = 450$	...	Cheap $a = 100$ $b = 100$ $c = 250$ $d = 300$
Distance to work place	Near $a = 0$ $b = 0$ $c = 15$ $d = 25$	Near $a = 0$ $b = 0$ $c = 15$ $d = 25$	Average $a = 15$ $b = 25$ $c = 30$ $d = 35$	Near $a = 0$ $b = 0$ $c = 15$ $d = 25$	Near $a = 0$ $b = 0$ $c = 15$ $d = 25$	...	Near $a = 0$ $b = 0$ $c = 15$ $d = 25$
Rental period	$x <= 6$ $a = 0$ $b = 0$ $c = 6$ $d = 6$	$x <= 6$ $a = 0$ $b = 0$ $c = 6$ $d = 6$	$x <= 6$ $a = 0$ $b = 0$ $c = 6$ $d = 6$	$6 < x <= 12$ $a = 6$ $b = 6$ $c = 12$ $d = 12$	$6 < x <= 12$ $a = 6$ $b = 6$ $c = 12$ $d = 12$	...	$x <= 6$ $a = 0$ $b = 0$ $c = 6$ $d = 6$
House condition	Very Good $a = b = c = d = 4$	Very Good $a = b = c = d = 4$	Good $a = b = c = d = 3$	Good $a = b = c = d = 3$	Very Good $a = b = c = d = 4$	...	Very Good $a = b = c = d = 4$

Table 10
Buyer's requirements

Variable	Variable type	Definition domain	Priority
Rental-rate (£)	Continuous (fuzzy)	{cheap(100,100,250,300), normal(250,300,400,450), expensive(400,450,600,600)}	20
Distance to work place (Minutes walk)	Continuous (fuzzy)	{near(0,0,15,25), average(15,25,30,35), far(30,35,50,50)}	10
Rental-period (months)	Continuous (crisp)	{$x \leqslant 6$(0,0,6,6), $6 < x \leqslant 12$(6,6,12,12), $x \geqslant 12$(12,12,24,24)}	5
House condition	Ranking (crisp)	{very-bad(0,0,0,0), bad(1,1,1,1), suitable(2,2,2,2), good(3,3,3,3), very-good(4,4,4,4)}	0.5

Table 11
Examples to be used by the inductive machine learning algorithm

House condition	Rental-period	Distance to *work place*	Rental-rate	Decision
1	$x \leqslant 6$	7	125	Purchase
1	$x \leqslant 6$	10	100	Purchase
1	$x \leqslant 6$	15	220	Purchase
2	$x \leqslant 6$	10	200	Purchase
3	$x \leqslant 6$	12	210	Purchase
4	$x \leqslant 6$	10	230	Purchase
1	$6 < x \leqslant 12$	5	180	Purchase
2	$6 < x \leqslant 12$	10	175	Purchase
3	$6 < x \leqslant 12$	17	200	Purchase
4	$6 < x \leqslant 12$	2	240	Purchase
3	$x \geqslant 12$	15	200	Purchase
3	$x \geqslant 12$	5	210	Purchase
2	$x \leqslant 6$	10	100	Purchase
1	$x \leqslant 6$	13	150	Purchase
4	$x \leqslant 6$	8	170	Purchase
1	$x \leqslant 6$	45	120	Purchase
3	$x \leqslant 6$	25	200	Purchase
2	$x \leqslant 6$	27	175	Purchase
4	$x \leqslant 6$	11	200	Purchase
4	$x \leqslant 6$	40	180	Purchase
1	$x \leqslant 6$	12	560	Do not purchase
3	$6 < x \leqslant 12$	40	120	Do not purchase
1	$6 < x \leqslant 12$	35	100	Do not purchase
2	$6 < x \leqslant 12$	37	200	Do not purchase
4	$6 < x \leqslant 12$	36	210	Do not purchase
1	$6 < x \leqslant 12$	28	220	Do not purchase
1	$x \geqslant 12$	37	110	Do not purchase
4	$x \leqslant 6$	7	310	Do not purchase
3	$x \leqslant 6$	27	470	Do not purchase
3	$6 < x \leqslant 12$	7	350	Do not Purchase

$(v_n, \ldots, v_2, v_1 : output)$, where v_i is the value that the input variable (attribute) V_i takes in the example, and the output is the decision that the buyer makes about the product (purchase, not purchase).

In our scenario, we show the buyer the accommodations $\{a_1, a_2, \ldots, a_{30}\}$ that have been taken randomly from the initial set $\mathscr{E}$. In Table 11, we show information about each accommodation and the decision that the buyer takes. Then, we apply the machine learning algorithm to the example set shown in Table 11 we obtain the following rules.

R_0: If distance to work place is `near` and *rental-rate* is `cheap` then rent the accommodation.
R_1: If *rental-period* is not more than 6 months and *rental-rate* is `cheap` then rent the accommodation.

R_2:   If *rental-rate* is `normal` or `expensive` then do not rent the accommodation.

R_3:   If *rental-period* is between 6 and 24 months and distance to the work place is `normal` or `far` then do not rent the accommodation.

From the above rules, we can obtain the constraint set C^f.

R_0^f: Distance to work place must be `near` and *rental-rate* must be `cheap`.

R_1^f: *Rental-period* must not be more than 6 months and *rental-rate* must be `cheap`.

R_2^f: *Rental-rate* must not be `normal` nor `expensive`.

R_3^f: *Rental-period* must not be between 6 and 24 months or distance to the work place must not be `normal` or `far`.

The priority associated with each constraint is then calculated by combining the priority associated with each variable (attribute) of the constraints and some measurement about the goodness of the rule from which the constraint is extracted. As a measure of the goodness of each rule, we employ the following fitness function taken from the genetic algorithm presented in Castro et al. (2001):

$$f(R_i) = \frac{e_p(e_{\text{tdc}} - e_n)}{e_{\text{tsc}} e_{\text{tdc}}}, \qquad (7)$$

where e_p is the number of examples in the training set, which are correctly classified, e_n is the number of examples which are erroneously classified, e_{tsc} is the total number of examples from the same class that exist in the training set, and e_{tdc} is the total number of examples from different classes that exist in the training set.

From this it can be seen that function (7) prefers rules that classify the largest number of examples incurring the lowest number of errors. For this example, the goodness associated with constraints R_0^f and R_1^f are as follows.

- $f(R_0^f) = 0.8$ ($e_p = 16$, $e_n = 0$, $e_{\text{tdc}} = 10$, $e_{\text{tsc}} = 20$).
- $f(R_1^f) = 0.7$ ($e_p = 14$, $e_n = 0$, $e_{\text{tdc}} = 10$, $e_{\text{tsc}} = 20$).

We then combine this information by means of the following formula:

$$\rho(R_i^f) = \max\{\rho(v_i) \in var(R_i^f)\} \times f(R_i), \qquad (8)$$

where $\rho(v_i)$ is the priority of the variable/attribute v_i, and $var(R_i^f)$ is the set of all variables of constraints R_i^f. Thus, in our example, the priorities associated with constraints R_0^f and R_1^f are 15 and 13, respectively.

4.3. Buyer's profile acquisition

In order to acquire the *buyer's profile model B*, the buyer is asked to fill in a registration form containing the restrictions and rewards that have been acquired from human sellers. In order to obtain the fuzzy truth associated with each proposition, that represents a restriction or reward, in the following form.

Table 12
Information associated with all products and buyer profiles

Likes or preferences information	Personal information
Phone	Male
Air-conditioner	Female
New-furniture	Native
Dislike-pet	Foreigner
Dislike-cooking	Student
District preferred	Professional

Table 13
A typical Buyer's profile

Proposition	Phone	Air-conditioner	New-furniture	Dislike-pet
Truth	0.6	0.2	0.1	0.9
Proposition	Dislike-cooking	District Bassett	Male	Foreigner
Truth	0.3	0.5	1	1

Notice the personal information that has a truth value of 0 is omitted from the table.

- When a proposition refers to the buyer's likes or preferences, the buyer estimates the fuzzy truth of this proposition by using a rating scale $(1, \ldots, n)$, and the fuzzy truth assigned to the proposition takes its normalized rating.
- When a proposition refers to personal information, its truth is 1 or 0.

In our example, the information that the registration form must hold is shown in Table 12 (according to Sections 4.1 and 4.2.). After the human buyer fills in the form, we can obtain the information about the buyer's profile model as shown in Table 13.

5. Why the fuzzy repertory table?

In this section we justify why the fuzzy repertory table technique is chosen to acquire the domain knowledge for the seller and buyer agents of our negotiation model.

Many manual and automatic methodologies have been developed for acquiring various kinds of domain knowledge for knowledge intensive systems. Some examples of such knowledge acquisition techniques are card or concept sorting (McDonald et al., 1986; Schreiber et al., 2000), the laddered grid (Shadbolt and Burton, 1989; Schreiber et al., 2000), interviewing (Gammack and Young, 1985; Evans, 1988; Graessar and Gordon, 1991; Schreiber et al., 2000), and the repertory grid technique. Among these techniques, we believe that our variant of the classical repertory grid, the fuzzy repertory table technique, is the most appropriate for acquiring the necessary domain knowledge from the seller and buyer agents' owners. Our rationale are given below.

The family of repertory grid techniques are useful for a number of reasons. (1) They have a solid foundation in human psychological theory—*Kelly's Personal Construct Theory* (Kelly, 1955). (2) Their utility in eliciting and acquiring knowledge from human users has been demonstrated (Boose and Bradshaw, 1987; Bradshaw et al., 1993; Gaines and Shaw, 1993, 1997). (3) They are relatively easy to formalize or implement, and so are easily employed by computer systems that aim to acquire knowledge in an automated way. (4) Little training is needed for users to master the technique (Schreiber et al., 2000). (5) Their dynamic and incremental nature makes them appealing to users. (6) They can analyse the acquired knowledge and thus reveal implied knowledge (e.g., the similarity or distinction between pieces of knowledge, hierarchical knowledge structures and entailment rules). When using other techniques, such as *protocol analysis* or *card sorting,* knowledge engineers have to analyse the acquired knowledge manually after the automated acquisition systems have done their jobs. This is usually a laborious process for knowledge engineers (Schreiber et al., 2000). (7) They enable us to obtain information about the attributes that can distinguish one product from others in a clear way. This is exactly what we want in developing our seller's and buyer's domain knowledge models.

Although many variation of the repertory grid exist (Cornisi, 1987), for our purposes they have the following limitations (Castro-Schez et al., 2003).

- *Classic* or *Kelly's repertory grid* (Kelly, 1955) with bipolar distinctions. If the technique is used for our purpose, users have to assign to each product one or the other pole of the distinction. This is not always possible because there are some attributes that could take intermediate states between left and right distinction poles. For instance, the attribute *accommodation-quality* can be rated between the left pole good and the right pole bad in our accommodation renting scenario.
- *Extended classic repertory grid* (Kelly, 1969). Here, users have to use values, chosen from a predefined rating scale, to rate each element against a given construct accurately. In other words, users have to make crisp distinctions. However, this is not always possible. For instance, users might not be able to rate the attribute *rental-rate* on a scale from 1 to 5 (where the minimum value 1 of this scale means `cheap` and the maximum value 5 means `expensive`).
- *Bathia's approach* (Bhatia and Yao, 1993). This approach allows users to rate an element by means of an interval rather than a crisp value. For instance, the user can assign the attribute *rental-rate* an interval value: [200,400]. However, the approach does not allow the user to rate elements by means of linguistic terms that are usually represented by trapezoid values instead of interval values.
- *Gaines' approach* (Gaines and Shaw, 1980). In this approach, the concept of a fuzzy set is introduced to represent each construct pole. When using this kind of repertory grid, for a given construct an assignment of a rating for an element determines the element's membership degree in the fuzzy set that defines each construct pole. For instance, the rating 2 for the attribute *rental-rate* is interpreted as membership degree 0.8 of the fuzzy set `cheap` (i.e., $\mu_{cheap}(2) = 0.8$), and it can also be interpreted as membership degree 0.2 of the fuzzy set `expensive` (i.e.,

$\mu_{expensive}$ (2) = 0.2). In other words, the attribute *rental-rate* is still rated by crisp values instead of fuzzy linguistic terms that we need.

- *Ford's approach* (Ford et al., 1991). The focus of the technique is on clarifying the logical rationale in the process of knowledge acquisition. By using this approach, users can obtain rules with uncertainty. However, when rating an element they also have to choose values from a predefined rating scale.

- *Hwang's approach* (Hwang, 1995). The approaches mentioned above do not represent linguistic information in an explicit way. Nevertheless, in real life people are often notoriously unwilling to give precise numerical rates, while they seem quite prepared to give qualitative ratings which usually are expressed by linguistic terms (Zadeh, 1975a, b). In order to address this issue, Hwang extends the repertory grid technique to the *fuzzy table*. In the fuzzy table, constructs are fuzzy attributes that can be rated by means of fuzzy linguistic terms from a finite set. For example, the attribute *rental-rate* can take the following fuzzy linguistic terms: `low`, `middle`, `high`, each of which is associated with a membership function. In this way, Hwang's fuzzy table is the most appropriate for coping with linguistic information. However, a repertory grid that only considers fuzzy linguistic terms is still inappropriate for our purposes because not everything is vague (for example, the attributes *sex* and *colour* are not vague).

Our FRT has none of the limitations mentioned above and thus is an appropriate method for our purposes. In more detail, the specific reasons for employing the FRT approach are as follows.

- FRT is suitable for processing, in a uniform manner, the most frequent attributes types that buyers and sellers use when they talk about a product during the course of a negotiation. In fact, the FRT allows the following types of attributes. (1) *Ordered-discrete or ordinal type*. For example, *district* where the accommodation is located, *accommodation-type* (e.g., `shared-house`, `flat`, `apartment`, `shared-room`, etc.). (2) *Unordered-discrete or nominal type*. For example, *accommodation-state* (`rented`, `available` and *reserve*), *buyer's-occupation* (`worker`, `student` and `unemployed`). (3) *Boolean type*. For example, *has-air-conditioner*?, *has-phone*?, *furnished*?, etc. (4) *Ranked type*. For example, *accommodation-quality* (the quality of a district where the accommodation is placed, etc). (5) *Continuous type*. For example, *rental-rate.*

- FRT allows interval and fuzzy linguistic terms to be assigned to continuous and ranked variables (attributes). For example, in FRT, the attribute *rental-rate* can be rated by the linguistic term cheap, and the attribute *rental-period* could be between 6 and 12 months. Also, it allows the expert to assign `any`, `several` or `no values` to one attribute. In this way, the vagueness inherent in many human judgements and preferences can be handled well by FRT.

- FRT employs trapezoidal functions (1) to represent any type of value (see Fig. 1). This representation has the following merits: (1) *adaptable:* it can be adapted to hold any values used by human users; (2) *linguistically based:* it is easy for human users to understand; (3) *mixed:* it can accommodate vague and crisp values at the

same time; and (4) *uniform:* it can be used by an inductive learning algorithm (Castro et al., 1999, 2001).

6. Conclusions and future work

In this paper, we have presented a process of knowledge acquisition that is designed to provide autonomous software agents with sufficient domain knowledge to negotiate on behalf of their human users using a particular bilateral multi-issue negotiation model.

The general class of repertory grid techniques was selected as being appropriate since they have a good history of use in KA contexts where human users have to express relationships between objects and any preferences that might hold over them. Moreover, repertory grids have one other important characteristic—they are capable of making explicit what is often implicit or tacit knowledge that a human might have towards objects in the world. Indeed, this was its original use in psychology when it was used to uncover the attitudes that subjects held about their social world.

A key innovation in this paper is the use of the Fuzzy Repertory Table for acquiring the necessary domain knowledge for negotiating agents. This was motivated primarily by a recognition that many judgements and preferences that we needed to capture could not easily be expressed in terms of numerical ratings or rankings. Rather, a more faithful representation would be able to allow the users to specify the linguistic terms and values that they would use in the domain. These linguistic terms may be realized as clear crisp values; however, in other cases the representation is best captured in terms of a function that captures the essential vagueness of the term.

In the expository example used in this paper it is clear how the KA steps proceed and how the knowledge acquired maps to the knowledge requirements of the negotiating model. In adopting the FRT it is clear that there is an initial overhead in terms of the amount of KA that is required to capture the semantics of the terms that are to be used in any particular domain. In an application area such as our accommodation renting scenario it is reasonable to suppose that general attributes and their possible values do exist across different types of potential buyer and seller. In fact it may be that the initial KA overhead is something that all domains will require in which agents operate automated negotiation. The important variations may be in the priorities that any individual attaches to the features of the products and their associated rewards.

There are, however, a number of issues that require further investigation.

- Despite progress in defining a KA process for our negotiating agents there is a need to evaluate empirically how effective the process is in acquiring the domain knowledge. This evaluation should take notice not only of the efficiency of the KA process but also the extent to which it accurately captures the knowledge of the various human participants (including whether this knowledge changes during the course of the negotiation encounter).

- We have only modelled the process of acquiring domain knowledge for our negotiating agents. As we analyse a wider class of agents, agent tasks and negotiation strategies we are likely to find the need to acquire knowledge that relates to the task and inference layers of CommonKADS. As we consider the knowledge requirements of a wider range of negotiating agents we will undoubtedly need to bring a wider range of KA techniques to bear on the problem of populating the knowledge bases of our agent proxies and configuring their behaviours.
- Various types of tradeoffs play a very important role in negotiations. However, this issue is not handled in this paper. Although we have studied the issue in the context of our negotiation model elsewhere (Luo et al., 2003c, d), it is still worth exploring further in the context of more widespread ranges of models.
- In many negotiations people typically have a best alternative to a negotiated agreement (BATNA). This represents the walk-away-point beyond which the individuals prefer no deal to the deal being offered. BATNA is not represented in our negotiation model. However, since it is very important our negotiation model should be extended in this direction and thus we need to study the associated knowledge acquisition processes.

Acknowledgements

References

Atkin, R., 1974. Mathematical Structure in Human Affairs. Heinemann, London.

Bhatia, S.K., Yao, Q., 1993. A new approach to knowledge acquisition by repertory grids. In: Bhargava, B., et al. (Eds.), CIKM 93: Proceedings of the Second International Conference on Information and Knowledge Management, Washington, DC, ACM, New York, pp. 738–740.

Boose, J.H., Bradshaw, J.M., 1987. Expertise transfer and complex problems: using AQUINAS as a knowledge acquisition workbench for knowledge-based systems. International Journal of Man—Machine Studies 26, 3–28.

Bradshaw, J.M., Ford, K.M., Adams-Webber, J.R., Boose, J.H., 1993. Beyond the repertory grid: new approaches to constructivist knowledge acquisition tool development. International Journal of Intelligent Systems 8 (2), 287–333.

Castro, J.L., Castro-Schez, J.J., Zurita, J.M., 1999. Learning maximal structure rules in fuzzy logic for knowledge acquisition in expert systems. Fuzzy Sets and Systems 101, 345–353.

Castro, J.L., Castro-Schez, J.J., Zurita, J.M., 2001. Use of a fuzzy machine learning technique in the knowledge acquisition process. Fuzzy Sets and Systems 123 (3), 307–320.

Castro-Schez, J.J., Castro, J.L., Zurita, J.M., 2003. Fuzzy repertory table, a method for acquiring knowledge about input variables to machine learning algorithms. IEEE Transactions on Fuzzy Systems 2003, accepted.

Cornisi, R. (Ed.), 1987. Concise Encyclopaedia of Psychology. Wiley-Interscience, New York.

Dubois, D., Prade, H., 1999. Qualitative possibility theory and its applications to constraint satisfaction and decision under uncertainty. International Journal of Intelligent Systems 14, 45–61.

Dubois, D., Fargier, H., Prade, H., 1994. Propagation and satisfaction of flexible constraints. In: Yager, R., Zadeh, L. (Eds.), Fuzzy Sets, Neural Networks and Soft Computing. pp. 166–187.

Dubois, D., Fargier, H., Prade, H., 1996. Possibility theory in constraint satisfaction problems: handling priority, preference and uncertainty. Applied Intelligence 6 (4), 287–309.

Evans, J., 1988. The knowledge elicitation problem: a psychological perspective. Behavior and Information Technology 1 (2), 111–130.

Ford, K.M., Petry, F.E., Adams-Webber, J.R., Chang, P.J., 1991. An approach to knowledge acquisition based on the structure of personal construct systems. IEEE Transactions on Knowledge Data Engineering 3 (1), 78–87.

Gaines, B.R., Shaw, M.L.G., 1980. New directions in the analysis and interactive elicitation of personal construct systems. International Journal Man–Machine Studies 13, 81–116.

Gaines, B.R., Shaw, M.L.G., 1986. Induction of inference rules for expert systems. Fuzzy Sets and Systems 18 (3), 315–328.

Gaines, B.R., Shaw, M.L.G., 1992. Integrated knowledge acquisition architectures. Journal of Intelligent Information Systems 1 (1), 9–34.

Gaines, B.R., Shaw, M.L.G., 1993. Basing knowledge acquisition tools in personal construct systems. Knowledge Engineering Review 8 (1), 49–85.

Gaines, B.R., Shaw, M.L.G., 1997. Knowledge acquisition, modeling and inference through the World Wide Web. International Journal of Human–Computer Studies 46 (6), 729–759.

Gammack, J., Young, R., 1985. Psychological techniques for eliciting expert knowledge. In: Bramer, M. (Ed.), Research and Development in Expert Systems. Cambridge University Press, London, pp. 105–112.

Graessar, A.C., Gordon, S.E., 1991. Questions answering and the organization of the world knowledge. In: Kesesen W., Ortony, A., Craik, F. (Eds.), Memories, Thoughts, and Emotions. Essays in Honor of George Mandler. Earlbaum, Hillsdale, NJ.

Hwang, G., 1995. Knowledge acquisition for fuzzy expert systems. International Journal of Intelligent Systems 10, 541–560.

Jennings, N.R., 2000. On agent-based software engineering. Artificial Intelligence 117 (2), 277–296.

Jennings, N.R., 2001. An agent-based approach for building complex software systems. Communications of the ACM 44 (4), 35–41.

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

Kelly, G.A., 1955. The Psychology of Personal Constructs. Norton, New York.

Kelly, G.A., 1969. A mathematical approach psychology. In: Maher, B. (Ed.), Clinical Psychology and Personality: The Selected Papers of George Kelly. Wiley, New York.

Luo, X., Leung, H.-F., Lee, J.H.-M., Jennings, N.R., 2003a. Prioritised fuzzy constraint satisfaction problems: axioms, instantiation and validation. Fuzzy Sets and Systems 136 (2), 155–188.

Luo, X., Jennings, N.R., Shadbolt, N.R., Leung, H.-F., Lee, J.H.-H., 2003b. A fuzzy constraint based model for bilateral, multi-issue negotiation in semi-competitive environments. Artificial Intelligence 148 (1-2), 53–102.

Luo, X., Jennings, N.R., Shadbolt, N., 2003c. Acquiring tradeoff preferences for automated negotiations: a case study. Proc. 5th International Workshop on Agent Mediated Electronic Commerce, Melbourne, 41–48.

Luo, X., Jennings, N.R., Shadbolt, N., 2003d. Knowledge-based acquisition of tradeoff preferences for negotiating agents. Proc. 5th International Conference on Electronic Commerce, Pittsburgh, 138–144.

McDonald, J., Dearholt, D., Paap, K., Schvanevedt, R., 1986. A formal interface design methodology based on user knowledge. CHI'86 Proceedings, pp. 285–290.

Rathod, P., 1981. Methods for the analysis of repertory grid data. Personal Construct Psychology. Recent Advances in the theory and practice. St. Martin's Press, New York.

Schreiber, G., Akkermans, H., Anjewierden, A., Hoog, R., Shadbolt, N., Van de Velde, W., Wielinga, B., 2000. Knowledge Engineering and Management: The CommonKADS Methodology. Massachusetts Institute of Technology Press, Cambridge, MA.

Shadbolt, N., Burton, A.M., 1989. The empirical study of knowledge elicitation techniques. SIGART Newsletter 108, 15–18.

Shaw, M.L., 1980. On Becoming a Personal Scientist: Interactive Computer Elicitation of Personal Models of the World. Academic Press, London.

Slater, P. (Ed.), 1977. Dimensions of Intrapersonal Space by Grid Technique, Vol 2: Dimensions of Intrapersonal Space. Wiley, London.

Zadeh, L.A., 1975a. The calculus of fuzzy restrictions. In: Zadeh, L.A., et al. (Ed.), Fuzzy Sets and Applications to Cognitive and Decision Making Processes. Academic Press, New York, pp. 1–39.

Zadeh, L.A., 1975b. The concept of a linguistic variable and its application to approximate reasoning. Part I. Information Science 8, 199–249; Part II. Information Science 8, 301–357; Part III. Information Science 9, 43–80.