Mixed-strategic Reasoning of the *i** Goal Model

Completed Research Paper

Sreenithya Sumesh **Aneesh Krishna**

Abstract

Goal-Oriented Requirements Engineering represents the stakeholder objectives using goals for making decisions about the choice of suitable non-functional requirements in view of goal models. In a competitive environment, stakeholders' may have conflicting goals. Therefore, there is a need for a goal analysis method which offers an alternative design option that achieves the conflicting goals of different inter-dependent actors in a goal model. To address circumstances where there is uncertainty, this paper proposes a game-theory based probabilistic, mixed-strategy approach to choose the best alternative to resolve the conflicting requirements issue. In this paper, a framework is proposed that applies Nash equilibrium based on multi-objective values for selecting an optimum strategy in the i goal model by considering the opposing goals reciprocally. By integrating Java with IBM ILOG CPLEX, the proposed method was developed and evaluated successfully using different case studies.*

Keywords: Goal model, Nash equilibrium, Mixed strategy, Optimization

Introduction

The success of any software system depends on the degree to which its requirements are met. The entire phenomenon of software development has increasingly become involved with Requirements Engineering (RE) as its crucial developmental aspect. Amongst the various periodic processes of Requirements Engineering, the beginning and the most important process is the elicitation of requirements. Other processes involved in Requirements Engineering are analysing, modelling, communicating, agreeing on and evolving requirements (Sommerville 2005). First, it is important to identify what are the goals or tasks for the system and keeping goals as the end result, determining what objectives need to be met by the system. This is possible by the process of elicitation. The elicitation process helps in determining the stakeholders and the information received from the stakeholders are analysed by the requirements analyst and goals are identified from the collected requirements. Goals are of two kinds – hardgoals and softgoals. Hardgoals are identified by stakeholders. These hardgoals determine the functions the system must perform. The job of the requirements analyst is to analyse the hardgoals to construct an improvised software system. The hardgoals are disintegrated into specific goals known as softgoals. There are specific goals that relate to the specific attributes such as accuracy, reliability, performance etc that is expected of the system. Alternative superior level options for system design is inspected by the requirements analyst in order to implement the desired system design (Franch et al. 2016).

Goals are used to model the requirements of the software system. This is performed by eliciting, elaborating, structuring, specifying, analysing, negotiating, documenting and modifying requirements. This method of creating a prototype of the software system's requirements is known are Goal-Oriented Requirements Engineering (GORE) (Sommerville 2005). Since the mid-nineties, the software engineering field has been predominantly working with goal models. Goals perform a crucial role in Goal Oriented Requirements Engineering and helps in deciphering the domain and deducing the intent of the stakeholders (Mylopoulos et al. 1999). Goals are developed at various levels of perception and understanding, as of strategic concerns to technical matters. Therefore, it is a very important creation during the early phases of RE (Franch et al. 2016; Karagiannis et al. 2016). Goals are created based on a multi-view representation that exhibits the way in which goals, actors, states, objects, tasks, and domain properties are connected in the given system (Van Lamsweerde 2004). Examples of largely used goal models are Knowledge Acquisition in Automated Space (KAOS) Model (Dardenne et al. 1991), *i** goal model (Yu and Mylopoulos 1995), Non-Functional Requirements (NFR) model (Yu and Mylopoulos 1995), Attributed Goal-Oriented Requirements Analysis (AGORA) Model (Kaiya et al. 2002), Tropos Model (Bresciani et al. 2004) and Goal-Oriented Requirements Language (GRL) (Amyot et al. 2010) Model. The *i** goal model is one of the most sought after and accepted goal models in the software engineering field. This is because it helps goal-oriented prototype of socio-technical systems and organisations. The *i** model is useful in creating prototypes of organisations and aids the essential processes of the socio-technical systems basing the structure on actors and their dependencies.

Goals can be conflicting in nature and each goal i.e. the requirement of a system may have various design options that can be opposing in nature. Therefore, in a realistic competitive scenario, goals of many stakeholders when conflicting in character, an analyst, during the requirements analysis stage, must work with several opposing goals of all actors that are inter-dependent in a goal model. The challenge to derive an optimal alternative design option for a goal model that has conflicting goals forms the nature of Requirements-based engineering. The issues faced by decision makers in the real world scenario, have to consider the inter-dependent relationships between actors. The real challenges faced in the real world have to be taken into consideration by creating a unique structure thus achieving multi-objective optimisation (Subramanian and Kaur 2015). This application of a realistic decision-making process allows to venture beyond analytical concepts, like the concept of game theory.

Game theory is a very useful decision making, inter-disciplinary tool that helps in finding optimal solutions in conflicting situations with the supposition that players are rational and behave according to their own interests (Kelly 2003). At first this theory was formulated for the areas of mathematics and economics. This theory provides mathematical solutions and is useful in problem analysis and deriving values of payoff matrices that represent the players' outcomes. This paper suggests a unique methodology that is structured on game theory for system exploration involving alternative design evaluation. In this paper, the game players are top softgoals that has conflicting natures and the game strategy is considered as the alternative design options of inter-dependent actors in the *i** goal model.

When research was conducted previously on game theory based goal analysis, it was performed without taking into consideration the inter-dependent relationship among the actors. This paper involves the inter-dependent actors in the i* model. The various opposing objectives is combined with their importance in order to arrive at a decision making based on game theory. A two person zero-sum game approach is adapted to the i* goal model. Multi-objective functions are decided upon to understand the importance of the i* goal model. This helps in deriving optimal alternative options of inter-dependent relationship among the actors based on each conflicting softgoal. Then the game theory is applied to assess alternative options for each actor according to each conflicting softgoal. In the final stage, an optimal solution is derived that involves a strategy in a situation of conflicting objectives. The implementation of the proposed approach is demonstrated by a case study. The next section provides an overview of the existing approaches, techniques and methods related to GORE and also the *i** model that are closely associated with our approach. This paper is structured as follows: Section 2 of this paper states the existing approaches, techniques and methods related to the *i** models that are closely connected to our proposed approach. Section 3 provides the methodology encompassing the different steps of our approach and a short introduction of the methods that are used in the study. Section 4 describes the case study used in this work. The end of the paper explains the conclusions drawn from this work.

Related Work

In present times, Requirement Engineering involves the use of goals in order to understand the reason behind the existence of the functionality in comparison to what the role of the functionality would be. Goals play an important role in aligning the organization's requirements along with its functionality. This section presents an overview of the existing methods in connection to the i* model that adapts our approach. Horkoff and Yu proposed a qualitative analysis method that is interactive and iterative for the i* goal models (Horkoff and Yu 2016). This method of goals analysis involved algorithms and tools. The challenge faced by this method is more than one goal having the same label thus leading to uncertainty in decision making. A multi-objective optimisation model was proposed by Heaven et al (Heaven and Letier 2011) for analysing alternative design options in the KAOS model. The multi-objective optimization model does not consider the non-functional requirements of the system. To overcome this NFR issue, Mairiza et al developed a Multi-Criteria Decision Analysis (MCDA) method and applied TOPSIS and MCDA for prioritising the alternative options (Mairiza et al 2014). In order to decide on alternative design options, an inter-actor quantitative goal analysis has been developed by Subramanian et al. (Subramanian and Gopalan 2015). There could arise ambiguity in the selection of numeric numbers and in order to avoid such ambiguity fuzzy numbers are used. The qualitative and quantitative goal analysis process for the i* goal and other models do not include goals with opposing functions. A systematic game theory method for deciding an optimal alternative design option for inter-dependent actors in the i* model by reciprocally balancing the multiple opposing objectives with their significance has not been developed in previous research studies. Fuzzy mathematical applications and a linear programming optimisation tool are useful tools for quantitative goal analysis to find an optimal strategy with opposing objective functions in the requirements-based engineering design and this tool is used in this study. This kind of study was first conducted by Subramanian et al. (Subramanian et al. 2018) where an alternative design option was found for each actor in the i* goal model with opposing objective functions. This did not address the actor's interdependency in relationships that is crucial for decision-making in a real world competitive scenario. This was the negative side of this proposal. The next section gives a brief introduction to the way that game theory and multi-objective optimisation method relates to our proposed approach.

Multi-objective Optimisation for Reasoning of Opposing Non-functional Requirements Based on Game theory

This study uses the concept of game theory in real world competitive scenarios to conduct the decision making process. The method is to combine multiple conflicting objectives based on their importance. This enables a specific decision making process. A mixed-strategic outlook is used to find out the alternative options of inter-dependent actors. By using the concept of game theory the opposing objectives are reciprocally balanced in the i* model. The concept helps in understanding the alternative options for each actor in relation to each opposing softgoal. The subsection given below provides a short introduction to the decision making process using game theory.

Game Theory involves multiple players with multiple strategies that gives an outcome for each strategic combination. Therefore, several people are involved in the decision making process. The process follows the concept that all players in the game are aware of the situation, strategies and selections of the opposing players. However, the complexity of real life situations makes it challenging to formulate most strategies (Law and Pan 2009). When a formal study of a game that involves interactive situations is conducted, optimal strategies for players are derived that helps in understanding the outcome of the game (Aplak et al. 2014). To find a solution to a game, the strategies can be grouped. This is called Nash Equilibrium. Grouping of strategies is crucial in Nash Equilibrium and solutions cannot be derived by singlehandedly deviating from the strategies. Nash Equilibrium is a steady game and at the point when the players cannot gain anything more by changing the strategies and this is called the saddle point in the Nash Equilibrium. In Nash Equilibrium, players cannot be changed, strategies cannot be altered thus making it a very steady game. In this paper, the mixed strategy Nash Equilibrium is adapted for easy calculation and representation. This helps in selecting an alternative in situations of conflicting goals. In the next section, a short introduction is given explaining how to optimise multi-objectives and how this helps decision-makers to find an optimal alternative option for achieving conflicting goals.

In real world scenarios, all environmental factors such as actors, goals, strategies, and decision-makers etc. need to be considered and evaluated based on the objectives. The effectiveness of the decision-making highly depends on this factor. Optimal strategies for opposing objectives have to be found by decision makers (Aplak et al. 2014). Operations research techniques must be applied. Best alternatives to be selected from a list of possible options to derive optimisation as opposed to single objective optimisation methods. Examples of operational research techniques are linear programming, non-linear programming and quadratic programming (Aplak et al. 2014; Mairiza et al 2014). The multi objective optimisation method gives rise to a set of solutions called Pareto solutions or Pareto frontier and the optimal value is chosen based on the Pareto frontier (Subramanian and Kaur 2015).

A multi-objective optimisation problem is represented mathematically as:

$$Max/Min\ [f_1(x), f_2(x), \dots \dots, f_n(x)\ ] \qquad (1)$$

where $f_1, f_2, \dots, f_n$ are scalar functions, x is an element of Y and Y is the set of constraints.

Formalization of Multi-objective Functions

A generalised complete structure of the i^* goal model is modelled by formalising the opposing objective functions in terms of softgoals, goals, tasks and resources. For formalisation, Strategic Rationale (SR) model is considered as a directed graph which is represented as G (N;R), where N represents the intentional elements such as goals, softgoals, resources and tasks that form a set of nodes and R represents the means-end, task decomposition, dependency and contribution links that form a set of edges of the graphs. The task of a decision-maker is to choose a cost-effective ideal alternative option from the choices. An objective function for each choice can be generated based on the elements of the graph. Given an i^* goal model, our aim is to select the best alternative option according to its impact on softgoals. Impacts are *Make; Help; Hurt; Break; Some-; Some+* which are represented as triangular fuzzy numbers that indicates the extent to which an alternative option fulfils the leaf softgoal. The impacts of the softgoal preferences are propagated to the top softgoals, to find the level of satisfaction or scores of top softgoals. Also, each of the leaf softgoals is assigned a weight ω based on its relative importance in achieving the goal.

Firstly, the scores for each top softgoal of each actor based on its inter-actor dependency under each alternative are calculated. For details on representing goals, weights, impacts and alternatives, readers are directed to Subramanian et al. (Subramanian and Gopalan 2015).

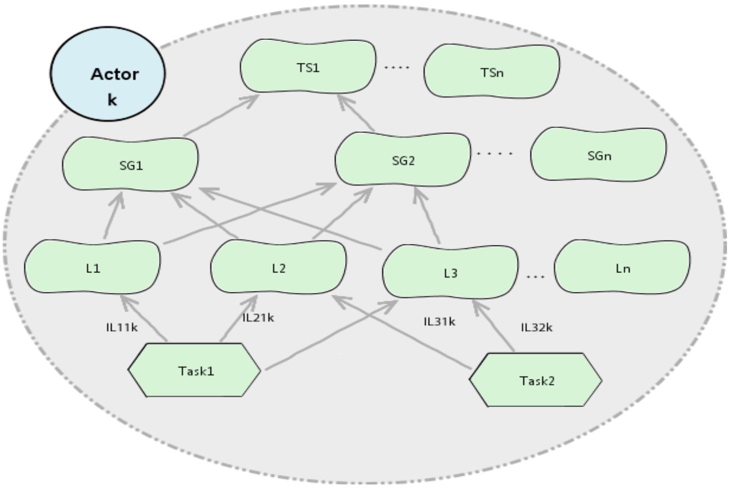

Figure 1 Directed Graph Representation

From Figure 1, consider the case of *t* hierarchy levels in the directed graph, with leaf softgoal at level zero. Let $\omega_{L_{ik}}$ represents the weight of i^{th} leaf softgoal and $I_{L_{ijk}}$means the impact on i^{th} leaf softgoal of j^{th} alternative of k^{th} actor. At level 1, if there are m number of softgoals, n_c children and n_d dependencies for the i^{th} softgoal, then the score of any softgoal at $t > 1$ is found by taking the product of its impact and each child score (Subramanian and Kaur 2016). Then the score of a softgoal at level *t* for an actor with a dependency relationship can be generalised as:

$$S_{SG_{i_tjk}} = \prod\nolimits_{l=1}^{m} I_{ijl} \sum_{i=1}^{m} \left\{ \sum_{d=1}^{n_c} \left[I_{d_{ij}} \times I_{d_{L_{ijk}}} \times \omega_{d_{L_{ijk}}} \right] + \sum_{y=1}^{n_c} \left[\sum_{b=1}^{n_d} \left(S_{i_{d_{by}}} \times I_{i_{d_{by}}} \right) \right] + \sum_{b=1}^{n_d} \left(S_{i_{d_b}} \times I_{i_{d_b}} \right) \right\} \tag{2}$$

Then the objective functions of top softgoals under each alternative for an actor are created from the scores as shown in Equation 2. If there is an inter-actor dependency relationship, then it is necessary to consider both strategic dependency and strategic rationale diagrams of the i^* goal model with the assumption that only softgoal inter-dependency relationships are taken into account in this approach. Consider that if there are *n* numbers of alternative options for an actor, then there are *n* objective functions for each top softgoal. To obtain a maximum score for the top softgoal under each alternative, the *n* objective functions that have to be maximised are given as:

$$f_{i(\omega_1)} = S_{SG_{i1k}} = Max \prod\nolimits_{l=1}^{m} I_{i1l} \sum_{i=1}^{m} \left\{ \sum_{d=1}^{n_c} \left[I_{d_{ij}} \times I_{d_{L_{i1k}}} \times \omega_{d_{L_{i1k}}} \right] \right. \\ \left. + \sum_{y=1}^{n_c} \left[\sum_{b=1}^{n_d} \left(S_{i_{d_{by}}} \times I_{i_{d_{by}}} \right) \right] + \sum_{b=1}^{n_d} \left(S_{i_{d_b}} \times I_{i_{d_b}} \right) \right\}$$

$$f_{i(\omega_2)} = S_{SG_{i1k}} = Max \prod\nolimits_{l=1}^{m} I_{i2l} \sum_{i=1}^{m} \left\{ \sum_{d=1}^{n_c} \left[I_{d_{ij}} \times I_{d_{L_{i2k}}} \times \omega_{d_{L_{i2k}}} \right] \right. \\ \left. + \sum_{y=1}^{n_c} \left[\sum_{b=1}^{n_d} \left(S_{i_{d_{by}}} \times I_{i_{d_{by}}} \right) \right] + \sum_{b=1}^{n_d} \left(S_{i_{d_b}} \times I_{i_{d_b}} \right) \right\}$$

…………………………………………………………………..

$$f_{i(\omega_n)} = S_{SG_{ink}} = Max \prod_{l=1}^{m} I_{i1l} \sum_{i=1}^{m} \left\{ \sum_{d=1}^{n_c} \left[I_{d_{in}} \times I_{d_{L_{ink}}} \times \omega_{d_{L_{ink}}} \right] + \sum_{y=1}^{n_c} \left[\sum_{b=1}^{n_d} \left(S_{i_{d_{by}}} \times I_{i_{d_{by}}} \right) \right] + \sum_{b=1}^{n_d} \left(S_{i_{d_b}} \times I_{i_{d_b}} \right) \right\}$$

$$\text{Such that } 0 \leq \omega_{d_{L_{jk}}} \leq 100 \; for \; d = 1 \; to \; n_c \qquad (3)$$

Similarly objective functions that have to be minimised are formalised for each actor in the i^* goal model. The next section explains how the multi-objective functions of opposing goals (maximum and minimum in nature) are optimised.

Evaluation of the Optimal Solutions of Multi-objective Optimisation Functions

In the proposed model, each actor is considered to have two opposing softgoals (SG_1 and SG_2) and two alternative design options (A_1 and A_2). Optimising the objective functions for softgoals (SG_1 and SG_2) individually can generate two ideal solutions using Algorithm 1. The IBM ILOG CPLEX optimisation tool is used for evaluating the optimisation process (Lima 2010). The IBM ILOG CPLEX optimizer is used to solve mathematical business models using powerful algorithms to obtain precise and logical decisions. Additionally, IBM ILOG CPLEX has a modelling layer called 'Concert' that enables interfacing with Java, C++ and C # languages.

Let the ideal solutions for the objective functions for softgoals (SG_1 and SG_2) of an actor using the two alternative design options (A_1 and A_2) based on the Equation 2 is expressed as

$$\left(x_{SG_1A_1}, x_{SG_1A_2}, x_{SG_2A_1}, x_{SG_2A_2} \right) \qquad (4)$$

Likewise, the multi-objective function values are generated for all the actors in the goal model. These optimal values refer to the capacity of each alternative to fulfil the stakeholders' objectives.

Mixed-strategy Equilibrium

Sometimes, decision-analysts should take a "mixed-strategy" approach to address uncertain circumstances. A framework is created in this paper that identifies the mixed-strategy Nash equilibrium based on the multi-objective values. The Nash equilibrium is useful for analysing the result of competitive scenarios, especially when applied to conflict situations. This paper proposes a probabilistic mixed-strategy approach to resolve the conflicting issue in choosing the best alternative by different actors for achieving the opposing goals. To demonstrate this approach given the space constraints, we considered a two-player (*X* and *Y*) game theory with an inter-actor dependency relationship from *X* to *Y*. Also, assume that both players have the same alternative options (A_1 and A_2) for reaching their opposing top softgoals. Based on finding the probability of each player, let *p* be the probability that *X* chooses A_1, so *(1-p)* is the probability that it chooses A_2. Similarly, let *q* be the probability that *Y* chooses A_1, so *(1-q)* is the probability that it chooses A_2. To find mixed-strategies, *p-mix* and *q-mix* options are computed. Then the optimal choice of each player when choosing from the various alternatives is found algebraically and graphically. To depict each player's choice of the mixing probability, a best response function for each actor is generated. The mixed-strategy Nash equilibrium is revealed by combining these best response functions. An intersection point can be discovered from the combined best response functions. At this point, the players "arrive" at a profile where every player's strategy is a best response to every player it self's. At that point, they will be in a "stable" situation called "Equilibrium". The final decision must cope with the decision context and characteristics i.e., supporting the decision-makers, criteria and the alternatives when addressing the different objectives. This step involves a pair-wise comparison between several alternatives to fulfil stakeholders' objectives. By using Nash's mixed-strategy equilibrium, the outputs provide an optimal selection of alternatives for each actor. Since inputs are optimal values from CPLEX, this final decision analysis leads to a Pareto optimal equilibrium.

Algorithm 1: Main Module- Optimal Selection

Input: A set of directed graphs $S = \{S_1, S_2, \dots, S_n\}$ such that *G* is a subset of S that have same *n* set of tasks T, where $G = \{G_1, G_2, \dots, G_k\}$. Each G_i is a quadruple $\{T, L, SG, TS\}$ where each element T, L, SG, TS represents a set of task, a set of leaf softgoals, a set of in-between softgoals, a set of top softgoals respectively with each top softgoal associated with opposing variables such as *Max* or *Min.*

$\boldsymbol{for}\ G_i \in G\ \boldsymbol{do}$
  $\boldsymbol{for}\ task\ t \in T\ \boldsymbol{do}$
    $\boldsymbol{for}\ top\ softgoals\ t_s \in TS\ \boldsymbol{do}$
      $\boldsymbol{if}\ t_s\ is\ Min\ \boldsymbol{then}$
        $Generate\ minimisation\ objective\ function$;
      $\boldsymbol{end}$
      $\boldsymbol{if}\ t_s\ is\ Max\ \boldsymbol{then}$
        $Generate\ maximisation\ objective\ function$;
      $\boldsymbol{else}$
        Break**;**
      $\boldsymbol{end}$
    $\boldsymbol{end}$
  $\boldsymbol{end}$
$\boldsymbol{end}$
$Let\ F_{Max} \leftarrow Max\{f_{max_1}, f_{max_2}, \dots \dots, f_{max_n}\}$;
$Let\ F_{Min} \leftarrow Min\{f_{min_1}, f_{min_2}, \dots \dots, f_{min_n}\}$;
$\boldsymbol{for}\ f_{max_i} \in F_{Max}\ \boldsymbol{do}$
  $Let\ x_{max_i} \leftarrow optimal(f_{max_i}, Max)$; //finding optimal solutions for maximum objective functions
$\boldsymbol{end}$
$\boldsymbol{for}\ f_{min_i} \in F_{Min}\ \boldsymbol{do}$
  $Let\ x_{min_i} \leftarrow optimal(f_{min_i}, Min)$; //finding optimal solutions for minimum objective functions
$\boldsymbol{end}$
$Generate\ each\ player's\ optimal\ best\ response\ function\ graph\ under\ different\ alternative\ options$;
$Combine\ the\ best\ response\ function\ graphs\ to\ obtain\ mixedstrategy\ Nash\ equilibrium\ point\ that\ represents\ the\ Pareto\ optimal\ choice\ for\ each\ player$;

Algorithm 2: Sub Module- Solving Multi-objective functions

ASSERTION: Solves the objective function to obtain the optimal function value:
Declare the variables;
Define the expressions, the objective functions and the constraints based on *C;*
$\boldsymbol{if}\ C\ is\ Max\ \boldsymbol{then}$
  Define maximisation function;
$\boldsymbol{end}$
$\boldsymbol{if}\ C\ is\ Min\ \boldsymbol{then}$
  Define minimisation function;
$\boldsymbol{else}$
$W \leftarrow cplex.solve()$;
$\boldsymbol{end}$
//invoking CPLEX function
return *W***;**

Telemedicine System Case Study

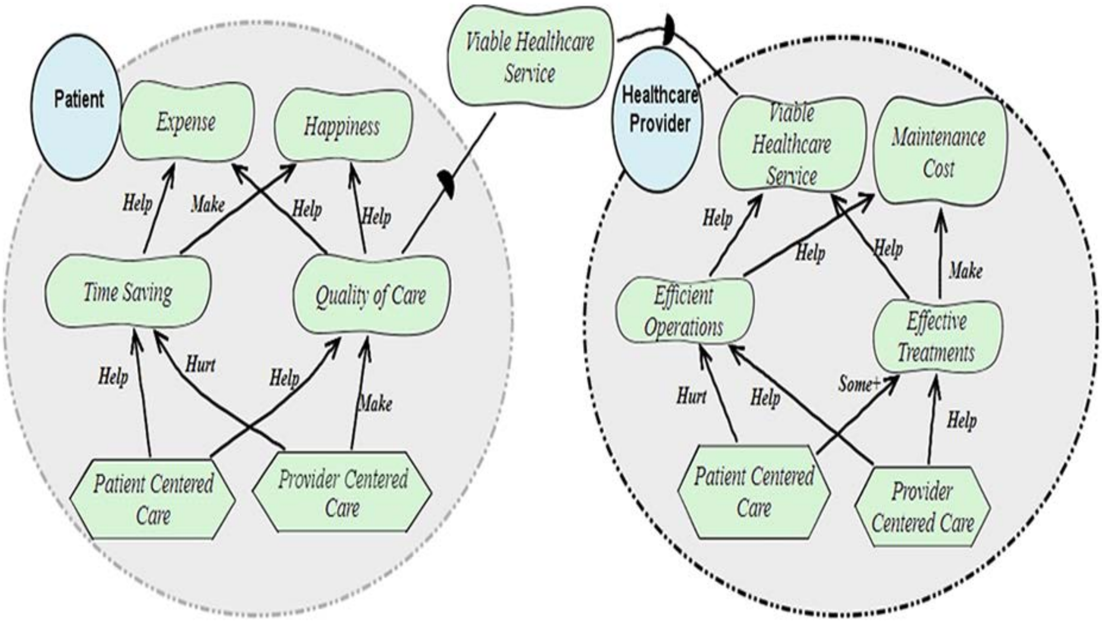

Figure 2 Telemedicine System

The mixed-strategy Nash equilibrium for the *i** framework was evaluated using the goal models from the existing RE literature: Meeting Scheduler System (van Lamsweerde 2004) and Telemedicine (Yu 2001). In order to illustrate the application of the proposed approach within the space restrictions, a generic telemedicine system case study is utilized (from the literature). The telemedicine system combines information technology and telecommunication to provide remote diagnosis and treatment for patients. The adapted telemedicine system (Figure 2) shows two actors, *Patient* and *Healthcare Provider* that are considerably simplified, but nevertheless require some kind of reasoning, namely the identification and exploration of alternatives. The main non-functional requirements or softgoals of the actor *Patient* are the *Expense* of the treatment and *Happiness* obtained from the remote treatment, which depend upon the softgoals *Time Saving* and *Quality of Care*. There are two alternative ways of obtaining treatment for the *Patient*. It is either via *Patient-Centered Care* or by *Provider-Centered Care*. The *Patient* has to choose an alternative option so that his/her *Expense* is less and *Happiness* is more.

The actor *Healthcare Provider* has two main non-functional requirements or softgoals namely *Viable Healthcare Service* and *Maintenance Cost* representing the *Healthcare Provider* aims of providing services in the telemedicine system. The goal *Viable Healthcare Service* can be implemented in one of two ways and thus is *OR* decomposed into two tasks: *Patient Centered Care* or *Provider Centered Care*. The selection of a task for this goal depends on the non-functional goals *Viable Healthcare Service* and *Maintenance Cost* for the satisfaction levels of actor *Healthcare Provider*. Here, the task is to select an alternative option that increases the *Viable Healthcare Service* and decreases *Maintenance Cost*. The objective of this system is to select the best alternative option based on its impact on each of the softgoals. The impacts indicate the extent to which an alternative option satisfies the corresponding softgoal. Impacts such as *Make, Help, Hurt, Break, Some+, Some-* are denoted by fuzzy triangular numbers. Along with the softgoal preferences, these impacts propagate to the top softgoals, to find the level of satisfaction or scores of top softgoals. For each actor, leaf softgoals are assigned an individual weight that can optimally select the best alternative option for achieving the opposing goals. For illustration and simplicity of calculation, de-fuzzification is used to convert the impacts which are represented in fuzzy numbers to quantifiable values (Chou et al. 2008).

These de-fuzzified values as shown in Table 1, and are used to evaluate the objective functions of each top softgoal.

Table 1 Impacts De-fuzzification

Impact	Fuzzy Contribution	De-fuzzified value
Make	(0.64, 0.8, 1)	0.8
Help	(0.48, 0.64, 0.80)	0.64
Some+	(0.32, 0.48, 0.64)	0.48
Some-	(0.16, 0.32, 0.48)	0.32
Hurt	(0, 0.16, 0.32)	0.16
Break	(0, 0, 0.16)	0

For actor *Patient*, the objective functions for both the top softgoals, *Expense* and *Happiness*, under both alternatives *Patient Centered Care* and *Provider Centered Care*, are found using Equation 2 and Equation 3, which are as follows:

$$F_{Expense}(\omega)_{Patient\ Centered\ Care} = Min(0.4096 \times \omega_1 + 0.4096 \times \omega_2 + 0.0524 \times \omega_3 + 0.1573 \times \omega_4)$$

$$F_{Expense}(\omega)_{Provider\ Centered\ Care} = Min(0.1024 \times \omega_1 + 0.512 \times \omega_2 + 0.2097 \times \omega_3 + 0.2097 \times \omega_4)$$

$$F_{Happiness}(\omega)_{Patient\ Centered\ Care} = Max(0.512 \times \omega_1 + 0.4096 \times \omega_2 + 0.0524 \times \omega_3 + 0.1573 \times \omega_4)$$

$$F_{Happiness}(\omega)_{Provider\ Centered\ Care} = Max(0.128 \times \omega_1 + 0.512 \times \omega_2 + 0.2097 \times \omega_3 + 0.2621 \times \omega_4)$$

Similarly, for the actor *Healthcare Provider*, the objective functions for both the top softgoals, *Viable Healthcare Service* and *Maintenance Cost*, under both alternatives *Patient Centered Care* and *Provider Centered Care*, can be generated using Equation 2 and Equation 3, based on their scores.

$$F_{Viable\ Healthcare\ service}(\omega)_{Patient\ Centered\ Care} = Max(0.1024 \times \omega_3 + 0.3072 \times \omega_4)$$

$$F_{Viable\ Healthcare\ service}(\omega)_{Provider\ Centered\ Care} = Max(0.4096 \times \omega_3 + 0.4096 \times \omega_4)$$

$$F_{Maintenance\ Cost}(\omega)_{Patient\ Centered\ Care} = Min(0.128 \times \omega_3 + 0.384 \times \omega_4)$$

$$F_{Maintenance\ Cost}(\omega)_{Provider\ Centered\ Care} = Min(0.512 \times \omega_3 + 0.512 \times \omega_4)$$

The solutions to these objective functions are obtained by invoking the IBM ILOG CPLEX. The obtained function values are given in Table 2 as ready reference. In studies of health care efficiency, the objective of production is perceived to be either providing services or achieving outcomes. By developing an appropriate probabilistic mixed-strategy approach, an optimal selection of the best alternative by different actors for achieving opposing goals can be achieved. Based on finding the probability of each player, *Healthcare Provider (HCP)* and *Patient*, *l*et p be the probability that *Healthcare Provider* chooses *Patient Centered Care* (A_1), so (1-p) is the probability that it chooses *Provider Centered Care* (A_2) and let q be the probability that *Patient* chooses A_1, so $(1-q)$ is the probability that it chooses A_2. To find mixed strategies, *p-mix* and *q-mix* options are computed. The *p-mix* and *q-mix* table is shown in Table 2. We can find *Patient* optimal choice of *A1* (q) in two ways: algebraically and graphically. In algebraically as shown in Table 3, *Patient* solves for the value of q that equates *HCP*'s payoff from choosing A_1 or A_2:

$$30.72q+40.96(1-q) = 12.8q+51.2(1-q)$$

$$q=0.36 \text{ i.e., } 36\%, \text{ so } 1-q=64\%$$

Graphically as shown in Figure 3, q is chosen so as to equalise the payoff that *HCP* receives from choosing both strategies. This requires the understanding about how the *HCP's* payoff varies with *Patient's* choice of q.

If *Patient* choose A_1 with q=36% and A_2 with 64%, then

a) *HCP* success rate of choosing A_1 is *30.72*0.36+40.96*(1-0.36) = 37%*

b) *HCP* success rate of choosing A_2 is *12.8*0.36+51.2*(1-0.36) = 37%*

Table 2 Objective Function Values

			p	*1-p*	
		Healthcare Provider			
	Patient		(*Happiness, Viable Healthcare service*)	(*Expense, Maintenance Cost*)	*p-mix*
q		*Patient Centered Care*	(51.2, 30.72)	(5.24, 12.8)	51.2**p* + 5.24*(1-*p*)
1-q		*Provider Centered Care*	(40.96, 40.96)	(10.24, 51.2)	40.96**p* + 10.24*(1-*p*)
		q-mix	30.72**q* + 40.96*(1-*q*)	12.8**q* + 51.2*(1-*q*)	

Since this is a constant sum game, *Patient's* success rate is *100% - HCP's* success rate i.e., *100-37=63%*. Similarly, we can algebraically find *HCP's* optimal choice of A_1 *(p)* as shown in Table 4. *HCP* solves for the value of *p* that equates *Patient's* payoff from choosing A_1 or A_2:

$$51.2p+5.24(1-p) = 40.96p+10.24(1-p)$$

i.e., p=0.33 i.e., 33%, so 1-p=67%

In graphically as shown in Figure 4, *p* has been chosen so as to equalise the payoff *Patient* receives from choosing both strategies. This requires the understanding about how the *Patient's* payoff varies with *HCP's* choice of *p*. If *HCP* choose A_1 with *p=33%* and A_2 with *67%*, then

a) *Patient* success rate of choosing A_1 is *51.2*0.33+5.24*(1-0.33) = 20%*

b) *Patient* success rate of choosing A_2 is *40.96p+10.24(1-p) = 20%*

Since this is a constant sum game, *HCP's* success rate is *100% - patient's* success rate,

i.e., *100-20=80%.*

Table 3 *HCF*'s Payoff Calculation Based On *q*

Patient's choice of *q*		*HCP* choosing A_1	*HCP* choosing A_2
0%	0	40.96	51.2
5%	0.05	40.45	49.28
10%	0.1	39.94	47.36
15%	0.15	39.42	45.44
20%	0.2	38.91	43.52
25%	0.25	38.40	41.6
30%	0.3	37.89	39.68
35%	0.35	37.38	37.76
40%	0.4	36.86	35.84
45%	0.45	36.35	33.92
50%	0.5	35.84	32
55%	0.55	35.33	30.08
60%	0.6	34.82	28.16
65%	0.65	34.30	26.24
67%	0.67	34.10	25.472
70%	0.7	33.79	24.32
75%	0.75	33.28	22.4

80%	0.8	32.77	20.48
85%	0.85	32.26	18.56
90%	0.9	31.74	16.64
95%	0.95	31.23	14.72
100%	1	30.72	12.8

Table 4 *Patient*'s Payoff calculation based on *p*

HCP's choice of *p*		*Patient* choosing A_1	*Patient* choosing A_2
0%	0	5.24	10.24
5%	0.05	7.54	11.78
10%	0.1	9.84	13.31
15%	0.15	12.13	14.85
20%	0.2	14.43	16.38
25%	0.25	16.73	17.92
30%	0.3	19.03	19.46
35%	0.35	21.33	20.99
40%	0.4	23.62	22.53
45%	0.45	25.92	24.06
50%	0.5	28.22	25.60
55%	0.55	30.52	27.14
60%	0.6	32.82	28.67
65%	0.65	35.11	30.21
67%	0.67	36.03	30.82
70%	0.7	37.41	31.74
75%	0.75	39.71	33.28
80%	0.8	42.01	34.82
85%	0.85	44.31	36.35
90%	0.9	46.60	37.89
95%	0.95	48.90	39.42
100%	1	51.20	40.96

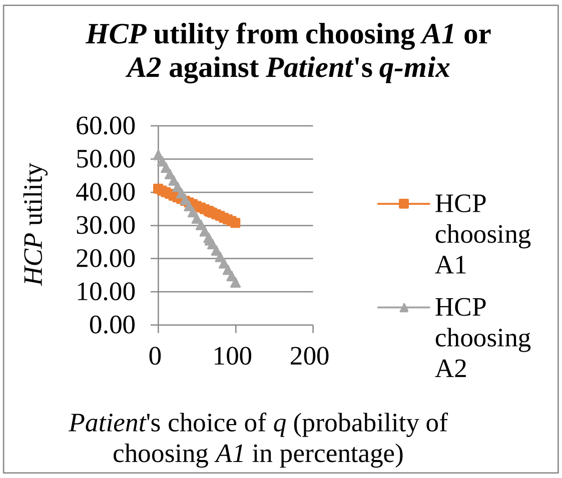

Figure 3 *HCP* utility against *q*

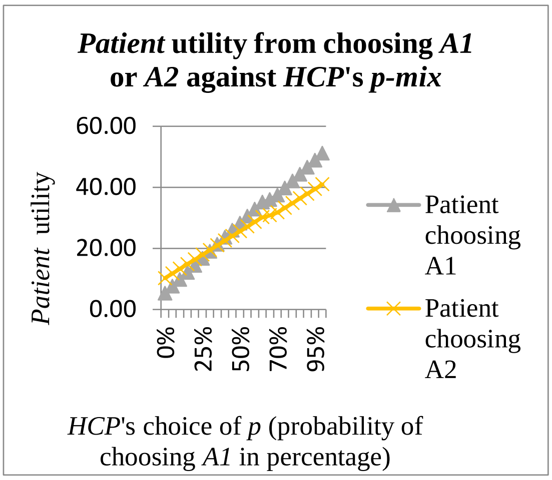

Figure 4 *Patient* utility against *p*

Generation of Best Response Function

Another way to depict each player's choice of the mixing probability is through best response function generation. (Recall p = probability (A_1) by *HCP*, q = probability (A_1) by *patient*). It shows from the Figures 5 and 6 that strategic best response of q = f (p) and p = g (q). If p, q $=0$ means, it always choose A_2; p, q $=1$ means, it always choose A_1. Combining the best response functions to obtain the intersection of two functions reveals the mixed-strategy Nash equilibrium. According to the Figure 7, the mixed-strategy Nash equilibrium is the point when *HCP* choose A_1 *33%* of the time (and A_2 *63%* of the time) while *Patient* choose A_1 *36%* of the time (and A_2 *64%* of time). The mixed-strategy Nash equilibrium is at point $(q, p) = (36, 33)$. At this point, the players "arrive" at a profile where every player's strategy is a best response to every player it self's. At that point, they will be in a "stable" situation called "Equilibrium". That's what a "Nash Equilibrium" is. Both choose the alternative *Patient Centered Care* with the probability of (36%, 33%) and choose alternative *Provider Centered Care* with a probability of (64%, 67%). That means there is a more chance for them to choose the strategy *Provider Centered care.* The Figure 7 indicates that the alternative *Provider Centered Care* (A_2) has a higher optimal value than the alternative *Patient Centered Care* (A_1). Hence by choosing the *Provider Centered Care* strategy, the system achieves the opposing top softgoals of inter-dependent actors in the i^* goal model reciprocally.

Conclusions

In this research a probabilistic mixed-strategic approach has been used for reasoning the non-functional requirements. This mixed-strategic approach of the Nash equilibrium-based goal analysis for the *i** goal model that has been proposed in this paper has helped resolve the conflict issue. This is achieved by choosing the best alternative by various actors for achieving the opposing goals. The proposed methodology is used in the Java Eclipse environment integrated with the IBM CPLEX tool. The optimal alternative selection method is used by balancing the opposing objectives of inter-dependent actors in the *i** goal model. Sensitivity Analysis is performed by way of further research. This provides useful input data to help the requirements analyst with the decision-making process.

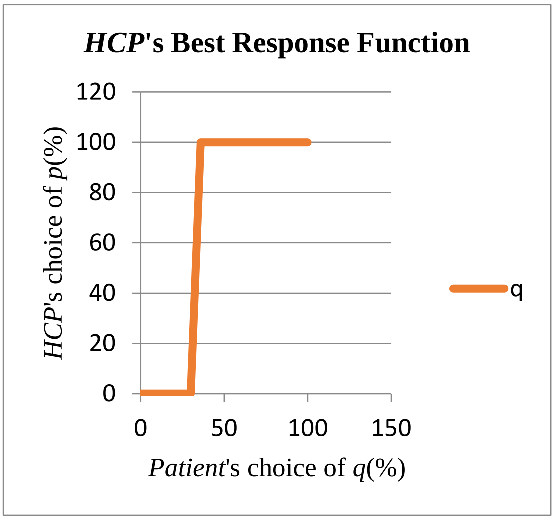

Figure 5 *HCP*'s Best Response Function

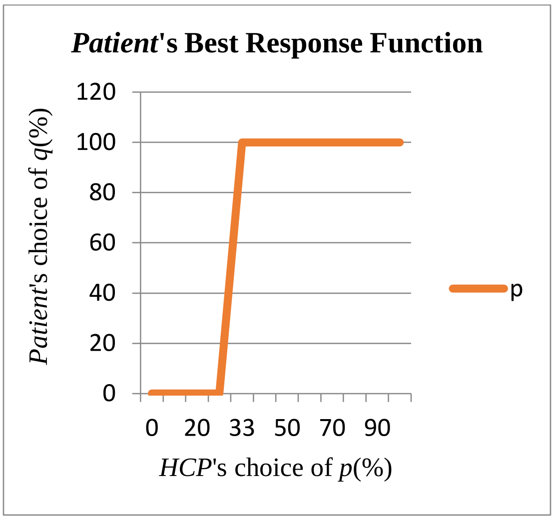

Figure 6 *Patient*'s Best Response Function

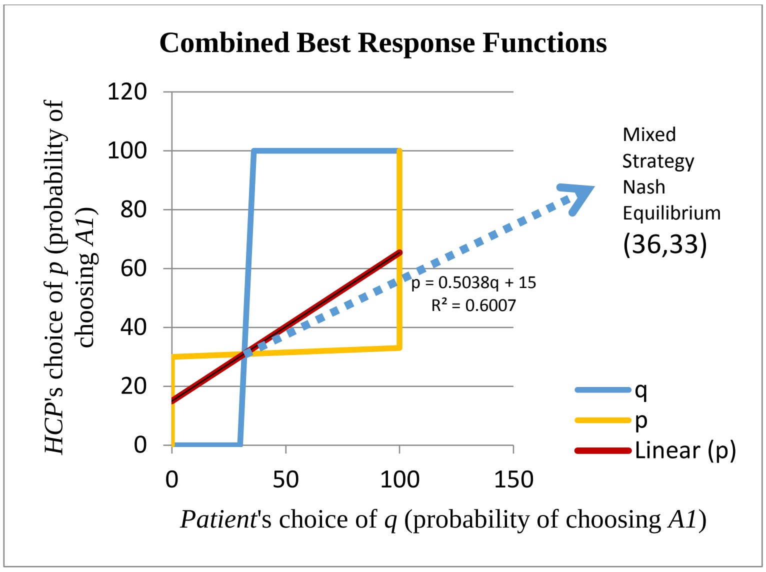

Figure 7 Combined Best Response Functions

References:

Amyot, D., Ghanavati, S., Horkoff, J., Mussbacher, G., Peyton, L., and Yu, E. 2010. "Evaluating Goal Models within the Goal-Oriented Requirement Language," *International Journal of Intelligent Systems* (25:8), pp. 841–877.

Aplak, H., Kabak, M., and Köse, E. 2014. "A Two Person Zero Sum Game Oriented to Integration of Objectives," *Journal of Military Studies* (5:2), pp. 65–85.

Bresciani, P., Perini, A., Giorgini, P., Giunchiglia, F., and Mylopoulos, J. 2004. "Tropos: An Agent-Oriented Software Development Methodology," *Autonomous Agents and Multi-Agent Systems* (8:3), pp. 203–236.

Chou, S.-Y., Chang, Y.-H., and Shen, C.-Y. 2008. "A Fuzzy Simple Additive Weighting System under Group Decision-Making for Facility Location Selection with Objective/Subjective Attributes," *European Journal of Operational Research* (189:1), pp. 132–145.

Dardenne, A., Fickas, S., and van Lamsweerde, A. 1991. "Goal-Directed Concept Acquisition in Requirements Elicitation," in *Proceedings of the 6th International Workshop on Software Specification and Design*, IEEE Computer Society Press, pp. 14–21.

Franch, X. 2006. "On the Quantitative Analysis of Agent-Oriented Models," in *CAiSE* (Vol. 6), Springer, pp. 495–509.

Franch, X., López, L., Cares, C., and Colomer, D. 2016. "The *i** Framework for Goal-Oriented Modeling," in *Domain-Specific Conceptual Modeling*, Springer, pp. 485–506.

Giorgini, P., Mylopoulos, J., Nicchiarelli, E., and Sebastiani, R. 2002. "Reasoning with Goal Models," *ER* (2), pp. 167–181.

Heaven, W., and Letier, E. 2011. "Simulating and Optimising Design Decisions in Quantitative Goal Models," in *Requirements Engineering Conference (RE), 2011 19th IEEE International*, IEEE, pp. 79–88.

Hillier, F. S. 2012. *Introduction to Operations Research*, Tata McGraw-Hill Education.

Horkoff, J., and Yu, E. 2011. "Analyzing Goal Models: Different Approaches and How to Choose among Them," in *Proceedings of the 2011 ACM Symposium on Applied Computing*, ACM, pp. 675–682.

Horkoff, J., and Yu, E. 2016. "Interactive Goal Model Analysis for Early Requirements Engineering," *Requirements Engineering* (21:1), pp. 29–61.

Kaiya, H., Horai, H., and Saeki, M. 2002. "AGORA: Attributed Goal-Oriented Requirements Analysis Method," in *Requirements Engineering, 2002. Proceedings. IEEE Joint International Conference On*, Ieee, pp. 13–22.

Karagiannis, D., Mayr, H. C., and Mylopoulos, J. 2016. *Domain-Specific Conceptual Modeling*, Springer.

Kelly, A. 2003. *Decision Making Using Game Theory: An Introduction for Managers*, Cambridge University Press.

van Lamsweerde, A. 2004. "Goal-Oriented Requirements Enginering: A Roundtrip from Research to Practice [Enginering Read Engineering]," in *Requirements Engineering Conference, 2004. Proceedings. 12th IEEE International*, IEEE, pp. 4–7.

Law, W.-W., and Pan, S.-Y. 2009. "Game Theory and Educational Policy: Private Education Legislation in China," *International Journal of Educational Development* (29:3), pp. 227–240.

Lima, R. 2010. "IBM ILOG CPLEX-What Is inside of the Box?," in *Proc. 2010 EWO Seminar*.

Mairiza, D., Zowghi, D., and Gervasi, V. 2014. "Utilizing TOPSIS: A Multi Criteria Decision Analysis Technique for Non-Functional Requirements Conflicts.," in *APRES*, pp. 31–44.

Mylopoulos, J., Chung, L., and Yu, E. 1999. "From Object-Oriented to Goal-Oriented Requirements Analysis," *Communications of the ACM* (42:1), pp. 31–37.

Sommerville, I. 2005. "Integrated Requirements Engineering: A Tutorial," *IEEE Software* (22:1), pp. 16–23.

Subramanian, C. M., Krishna, A., and Gopalan, R. P. 2015. "Quantitative Reasoning of Goal Satisfaction in the *i** Framework.," in *SEKE*, pp. 666–669.

Subramanian, C. M., Krishna, A., and Kaur, A. 2015. "Optimal Reasoning of Goals in the *i** Framework," in *Software Engineering Conference (APSEC), 2015 Asia-Pacific*, IEEE, pp. 346–353

Subramanian, C. M., Krishna, A., and Kaur, A. 2016. "Optimal Goal Programming of Softgoals in Goal-Oriented Requirements Engineering.," in *PACIS*, p. 202.

Subramanian, C. M., Krishna, A., and Kaur, A. 2018. "Game Theory-Based Requirements Analysis in the *i** Framework," *The Computer Journal* (61:3), pp. 427–446. (https://doi.org/10.1093/comjnl/bxx110).

Yazdania, S., Linb, Y.-T., Caic, W., and Huanga, E. 2017. "A Game Theory Perspective on Requirement-Based Engineering Design, 15th Annual Conference on System Engineering Research," *Disciplinary Convergence: Implications for Systems Engineering Research*.

Yu, E. 2001. "Agent-Oriented Modelling Software versus the World," in *International Workshop on Agent-Oriented Software Engineering*, Springer, pp. 206–225.

Yu, E. S., and Mylopoulos, J. 1995. "FROM ER TO 'AR'—MODELLING STRATEGIC ACTOR RELATIONSHIPS FOR BUSINESS PROCESS REENGINEERING," *International Journal of Cooperative Information Systems* (4:02n03), pp. 125–144.