Interest-based Negotiation for Policy-regulated Asset Sharing

Christos Parizas[1], Geeth De Mel[2], Alun D. Preece[1], Murat Sensoy[3]
Seraphin B. Calo[2] and Tien Pham[4]

[1] School of Computer Science and Informatics, Cardiff University, UK
{C.Parizas, A.D.Preece}@cs.cardiff.ac.uk
[2] IBM T.J. Watson Research Center, Yorktown Heights, NY, USA
{grdemel, scalo}@us.ibm.com
[3] Department of Computer Science, Ozyegin University, Istanbul, Turkey
murat.sensoy@ozyegin.edu.tr
[4] US Army Research Lab, Sensors & Electron Devices Directorate,
Adelphi MD, USA
tien.pham1.civ@mail.mil

Abstract. Resources sharing is an important but complex problem to be solved. The problem is exacerbated in a coalition context due to policy constraints, that reflect concerns regarding security, privacy and performance to name a few, placed on the resources. Thus, to effectively share resources, members of a coalition need to negotiate on policies and at times refine them to meet the needs of the operating environment. Towards achieving this goal, in this work we propose and evaluate a novel policy negotiation mechanism based on the interest-based negotiation paradigm. Interest-based negotiation, promotes collaboration when compared with the traditional, position-based negotiation approaches.

1 Introduction

Negotiation is a form of interaction usually expressed as a dialogue between two or more parties with conflicting interests that try to achieve mutual agreement about the exchange of scarce resources, resolve points of difference and craft outcomes that satisfy various interests. Chasing mutual agreements, the involved parties make proposals, trade options and offer concessions. The automation of the negotiation process and its integration with autonomic, multi-agent environments has been well-researched over the last few decades [1, 2].

The theoretical approaches for automated negotiation can be classified into three major categories: (1) game theoretic (2) heuristic, and (3) argumentation based [1]. The first two represent traditional, bilateral negotiation mechanisms wherein each negotiation party exchanges offers aiming to usually satisfy their own interests. Both approaches fall under the broader spectrum of position-based negotiations (PBN), where participants attack the opposing parties' offers, trying to convince them for the suitability of their own ones. Typically, those approaches are formalized as search problems in the space of possible deals by focusing on negotiation objectives.

Argumentation-based negotiation (ABN) has been introduced as a means to enhance automated negotiation by exchanging richer information between negotiators. Interest-based negotiation (IBN) is a type of ABN based on a mechanism, where negotiating agents exchange information about the goals that motivate the negotiation action [3,4]. IBN unlike PBN approaches, tackles the problem of negotiation, focusing on "why to negotiate for" rather than on "what to negotiate for", aiming to lead negotiating parties to win-win solutions.

Multi-party teams are often formed to support collective endeavors, which otherwise would be difficult, if not impossible, to achieve by a single party. In order to support such activities, resources belonging to collaborative partners are shared among the team members. Mechanisms, for effective resource sharing between institutions and/or individuals are actively and broadly explored in research community. This is due to the impact that different resource sharing setups and modifications (what to share, with who, when and under what conditions) can bring into the collaboration, with respect to domains such as security, privacy and performance to name only a few.

Consider for instance the following scenarios: a) the resource sharing in corporate environments such as the recent *MobileFirst* partnership between IBM and Apple where cloud and other services are shared in a daily basis; or b) a short-lived, mobile, opportunistic network comprised of few peer members, established for message routing or data sharing. In both cases, an access control mechanism that governs resource sharing, needs to be implemented for establishing smooth collaboration. A suitable mechanism for managing access control on resources of such systems is a policy-based management system (PBMS). A PBMS provides systems administrators with a programable, abstract layer that describes the system to be managed, enabling them to express high-level, management goals and objectives through high-level policy rules.

The more complex and heterogeneous a multi-party, collaborative formations is, the more complex the mechanism that establishes trust between collaborators is as well; this has a negative impact on developing stricter resource sharing policy rules, which raises the barriers towards smooth and effective collaboration. In such scenarios, the need for a tool for enabling authorization policy negotiation is imperative, in order for strict policy rules to be refined accordingly, so that to promote collaboration.

The work herein presents a novel, interest-based policy negotiation mechanism for enabling authorization policy negotiation in multi-party, collaborative and dynamic environments. It focuses on policy makers who are not necessarily experts in either IT or negotiation techniques. To the best of our knowledge there is no mature work done on policy negotiation in general. The vast majority of automated negotiation work: a) deals with autonomous, multi-agent environments, b) utilizes PBN approaches and c) invariably ignores the special characteristics of multi-party, collaborative environments.

It is our belief that by understanding the interests behind collaborative parties' policies and by crafting options that can meet their asset sharing requirements, IBN could provide a negotiation mechanism, that promotes good collaboration unlike PBN, which inadvertently creates adversarial negotiation atmosphere. Moreover, the PBN paradigm with its fixed, opposing positions is a cumbersome negotiation method, to cope with more dynamic environments [1]. From an architectural point of view, the proposed negotiation mechanism can operate in parallel to a PBMS. Briefly, the proposed, policy IBN mechanism considers an approach that refines strict policies, in order to increase overall usability of collaborators' assets while remaining faithful to existing authorization policies. The main contributions of this of work are as follows:

1. Definition of an interest-based authorization policy negotiation model.
2. Specification of an architecture for its integration with PBMS.
3. Evaluation of policy IBN behavior through simulation experiments.

The remainder of the paper is organized as follows: in Section 2 we discuss previous literature on policy negotiation approaches. Section 3 presents an illustrative walkthrough of the policy negotiation mechanism. Section 4 describes the policy negotiation framework, the policy language, and its interface to PBMS by means of an architectural overview. Section 5 presents the algorithmic steps for IBN achievement through policy refinement and in Section 6 we evaluate IBN through simulation of multi-party, collaborative environments. We conclude this document in Section 7 by summarizing our contribution and outlining future research directions.

2 Background & Related Work

The first computer applications for supporting bilateral negotiations were developed in late 1960s [5]. The reason for their emergence was to assist human negotiators to overcome weaknesses related to negotiation process such as cognitive biases, emotional risks, and their inability to manage complex negotiation environments. Although there is rich literature on negotiation protocols in autonomous, Multi-agent Systems (MAS), there is very limited and no mature work done on policy negotiation.

Briefly, an agent in the context of MAS, is perceived as a software computational entity, capable of possessing the properties of autonomy, social ability, reactivity and pro-activeness [6]. In order for MAS agents to cooperatively solve problems, a comprehensive interaction is needed. Negotiation is an effective agent interaction mechanism, enabling autonomous bidirectional deliberation in both situations of competition and cooperation. For the development of sophisticated, negotiation models there are three areas that need to be considered: a) the negotiation protocols that define the rules of interaction amongst agents, b) the negotiation objects that contain the range of issues on which agreements must be achieved and c) the negotiation decision making models that guide agents' concession stance [7].

Initially, automated negotiation has received considerable attention in the field of economics, utilizing the analytical methods of game theory [8], aiming to calculate the equilibrium outcome before the negotiation game is played. While interesting conceptually, the game theoretic approaches have been criticized for assuming: a) complete and common information and b) perfect and correct information. However, most real world problems are cases of imperfect, erroneous and incomplete information where revelation is not realistic [4].

Heuristic negotiation approaches, started to being studied to cope with the computationally expensive game theoretic ones, which unrealistically considered the agents as entities of unlimited computational resources and time. Thus, they focused on producing good, but efficient negotiation decisions, as opposed to the optimal and inefficient decisions provided by predecessors [9]. The two basic limitations of heuristic approaches were: a) the underused agent communication and cognitive capabilities (e.g., agents' rejection as a feedback when a negotiation agreement is not achieved) and b) the statically defined agents positions (i.e., each agent has a clearly defined and static position) [4].

The intuition behind ABN is that the negotiating parties can improve the way they negotiate by exchanging explicit information about their intentions. This information exchange reveals unknown, non-shared, incomplete, and imprecise information about the underlying attitudes of the parties involved in the negotiation [10]. Think for instance, a negotiation case where two negotiators after exchanging offers are very close to achieve an agreement, but lacking this extra information they give up moments before achieving it.

We see the role of PBMS in managing large, complex and dynamic systems as of a high importance and the existence of sophisticated ways to do so imperative. We believe that the integration of an effective negotiation mechanism on a PBMS, works towards this direction.

To the best of our knowledge, no work had previously attempted to bring the IBN paradigm into policy negotiation. The authors in [11] focus on the requirements of policy languages, which deal with trust negotiation and pay attention on the technical aspects and properties of trust models to effectively negotiate with access requests. They do not research any of the aspects of policy negotiation and the scenarios they deal with are less dynamic compared to our problem domain.

The work in [12] proposes an architecture that combines a policy-based management mechanism for evaluating privacy policy rules with a policy negotiation roadmap. It is very generic and does not provide clear evidence of any effectiveness of the proposed approach, while lacking any evaluation. [13] is the first work that looks into policy negotiation and covers the area in depth. It also looks into collaborative environments and introduces the notion of ABN in policy negotiation. However, it focuses on a very specific application domain in which it deals with writing insurance policies. The whole process is based on a static approach maintaining a common and collaborative knowledge base.

The work discussed in [14] has many similarities to our work; it deals with cooperative environments where a PBMS is employed for managing the service composition in a distributed setting, while a negotiation framework is used to effectively compose services. The main difference with the work proposed herein is that the objective of the negotiation in [14] is the services that are managed by policies, not the policies themselves. We believe that in order to decrease the management overhead, the objective of the negotiation should be the policies. This is because policies, are the core of a PBMS and the logical component where the system's management resides.

Finally, [15] proposes a policy negotiation approach and presents its architecture. It lacks of any effectiveness evaluation while it does not consider either multi-partner or dynamic environments, following the PBN paradigm.

3 IBN and Asset Sharing Policies: Setting the Scene

Below, we provide illustrative scenarios to explain and motivate the use of IBN on policies, in charge of regulating asset sharing in collaborative environments. In Subsection 3.1 we revisit the classic orange–chefs scenario discussed in best-selling book *Getting to YES* [3] and then we transfer the same IBN principles into an opportunistic, mobile asset sharing scenario in Subsection 3.2.

3.1 The Chefs-Orange Scenario

Two chefs that work in the same kitchen, both want to use orange for their recipes. Unfortunately, there is only one orange left. Instead of negotiating on who is going to get the orange or some portion of it, as in a zero-sum, PBN approach, the two chefs opt to follow an IBN inspired approach. So, they ask each other why they need the orange for. In other words, they try to better understand their underlying goals of using the orange. Answering the "why" question, it turns out that one chef needs only the orange flesh (to execute a sauce recipe) while the other needs only its peel (for executing a dessert recipe) leading them to share the orange accordingly, achieving a win-win negotiation outcome.

3.2 Asset Sharing Policy Negotiation

An individual P2 wants to access a smartphone device SMD, owned by the individual P1. However, P1 has a set of restrictions which are captured by policy set R on how to share SMD with other people. These restrictions may reflect privacy concerns (e.g., by accessing their smartphone, one could have access to their photos). For the sake of clarity, in this example, we assume that the set R contains the following policy constraint R1: *do not share the device SMD with anyone else but its owner*. When P2 asks for permission to use the physical device SMD, R1 prohibits this action. Ostensibly there is little room for negotiation here with the current set of policies, if one follows a PBN approach.

However, by applying the IBN and trying to understand the underlying interests of the involving parties, we believe the situation could be handled in a satisfactory manner for both parties. For example, asking the "why" question it turns out that P2 needs a data service (as opposed to the physical device), in order to execute the task of *Email submission* and P1 does not mind sharing a data connection as a hotspot with a trusted party; if P1 could get to know why P2 needs the device for, the situation could be solved to the satisfaction of both parties. All an IBN mechanism needs to do in this case is to introduce another policy – actually a refinement of the existing policy – to R1 to say that data service can be shared among trusted parties. We argue that in such cases, by understanding the situation and broadening the space of possible negotiation deals, one can reach a win-win solution.

As stated earlier, IBN is a type of ABN where the negotiating parties exchange information about their negotiation goals, which then guide the negotiation process. Thus, the why party of the intention is of major importance when compared with the what part. We would say, that the IBN is more of a negotiation shortcut method rather than a typical negotiation process. By attacking the problem of negotiation, IBN could potentially skip the proposals making, the options trading and the need for negotiating parties to offer concession as in PBN cases. Instead of trying to negotiate on a fixed pie, it tries to find alternatives so that to expand it. In the next section, we shall introduce our IBN-based policy mechanism and provide our intuition behind the approach.

4 Interest-based Policy Negotiation Mechanism

The designing and development of intelligent tools and protocols for enhancing the negotiation process amongst human negotiators, needs to achieve some desirable outcomes that are secured by meeting a set of systematic properties such as: guaranteed negotiation success (i.e., negotiation mechanism that guarantees agreement), simplicity (i.e., eases negotiation decision for the participants), maximization of social welfare (i.e., maximization of the sum of payoffs or utilities of participants) to name a few. A complete list of desirable negotiation outcomes and evaluation criteria as described throughout the literature can be found in [16]. The main objective of the negotiation mechanism proposed herein, is to increase of social welfare.

In scenarios that often suffer from resource scarcity (i.e., environments where resource demand exceeds supply), and many user tasks may be competing for the same resource in order to be served, like those described in introduction, the formation of coalitions offers alleviation by bringing more resources on the table. The relationships between collaborative parties in those scenarios are mostly peer-to-peer (P2P), without assuming fully cooperative relationships. Coalition partners often pursue cooperation but they deny to share sensitive intelligence that can deliver greater value to the collaborators [17]. In literature this kind of relationship model, where parties have cooperative and competitive attitudes from time to time, is called coopetition [18]. The PBMS is in charge here, playing a regulative role in order to keep balance between asset sharing and asset "protection".

The mechanism presented herein allows negotiation on policies with minimal human intervention. In traditional system management, policies associated with PBMS are static (or rarely change); these systems, however, fail miserably in dynamic environments where policies need to adopt according to situational changes. We note, that it is not prudent to assume human operators in these environments that can effectively be on top of every change to manage PBMS(s) effectively; they require automated assistance.

Summarizing the intention behind applying IBN principles, on policy regulated asset sharing, it considers a cooperative negotiation approach, for strict policies refinement, that aims to: a) increase social welfare by increasing the overall usability of collaborative assets while b) remaining faithful to existing authorization policies, maintaining their core trends. Utilizing such a tool, a multilateral policy transformation can be achieved establishing a more effective PBMS, considering input and criteria from multi-party formations, for the benefit of the coalition.

The product of IBN execution is a new, refined, authorization policy rule. The IBN mechanism when refining the strict policy, considers the interest of both: asset owner and asset requestor. As far as the negotiation protocol is concerned, each negotiation session considers sets of two negotiators, so we deal with a bilateral negotiation mechanism. The issue that needs to be settled through the negotiation process, is the granting (or not) of access to non-sharable assets through policy refinement, thus, the protocol deals with single-attribute negotiations.

4.1 Policies Under Negotiation

The proposed policy negotiation framework is applied on authorization policies expressed in the Controlled English (CE) policy language [19]. CE policy language is an ontological approach that uses a Controlled Natural Language (CNL) for defining a policy representation that is both human-friendly (CNL representation) and unambiguous for computers (using a CE reasoner) [20]. CE is used to define domain models that describe the system to be managed. The domain models take the form of concept definitions and comprise objects, their properties, and the relationships amongst them. Those domain model components are the building blocks of the attribute-based CE policy language.

Each policy rule follows the if-condition(s)-then-action form and consists of four basic grammatical blocks as shown below:

- **Subject:** specifies the entities (human/machine) which interpret obligation policies or can access assets in authorization policies
- **Action:** what must be performed for obligations and what is permitted for authorization
- **Target:** objects on which actions are to be performed
- **Constraints:** boolean conditions

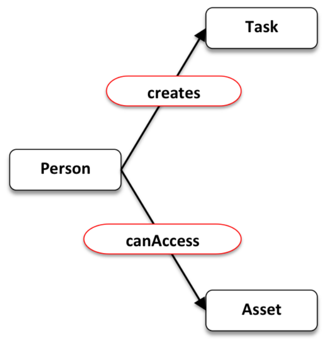

Fig. 1. Authorization Policy Negotiation Scenario: Domain Model.

The utilization of CE here is two-folde. It is not only the user friendly formal representation of a) the system to be managed and b) its policy-based management, but it also helps decision makers who lack technical expertise to cope in a more transparent way with the complexities associated with policy negotiation. It does so by easing the comparison between the original and the refined policy proposed by IBN, using a user friendly representation. Figure 1 provides a graphical depiction of the CE-based domain model, which describes the smartphone access scenario of Section 3.2 , while the CE representation of policy R1 is shown below.

```
Policy R1
If
( there is an Asset A named SMD ) and
( there is a Person P named P1 )
then
( the Person P canAccess the Asset A )
.
```

4.2 IBN Integration into Policy Regulated Asset Sharing

The role of policies in managing a system, is to guide its actions towards behaviors that would secure optimal system's outcomes. Different users have different rights, relationships and interests in regards to deployed coalition assets. Non-owner users want to gain access to assets in order to increase the probability of serving their tasks' needs, while owners want to protect their assets from unauthorized users. There is therefore a monopolistic resource usage case. The proposed negotiation approach considers both concerns in a single mechanism providing a mechanism that pursues a win-win negotiation outcome for any sets of negotiators. In other words, it tries through negotiation to redefine what is a suboptimal system outcome given: a) the currently-deployed assets, b) the user created tasks' needs and c) the policies themselves.

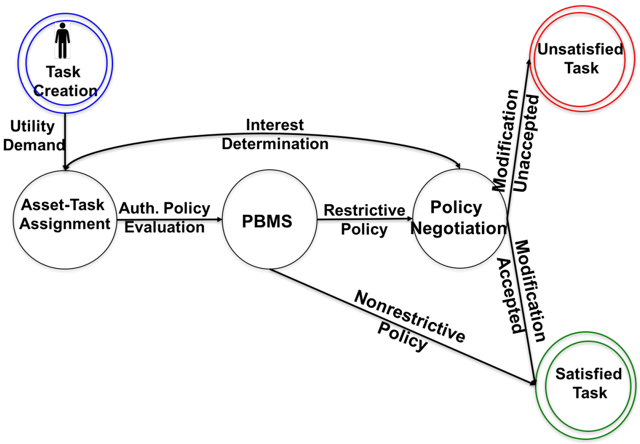

Fig. 2. Interest-based policy negotiation and task implementation.

The finite state diagram of Figure 2 depicts the role the policy negotiation mechanism plays on tasks' implementation in collective endeavors. The human, task creator, in order to serve their appetite for information, creates tasks with a utility demand. The Asset-Task Assignment (ATA) component is in charge for optimizing the task utility by allocating the appropriate assets (information-providing assets) to each task. The PBMS component is responsible then, for evaluating and enforcing authorization policies made by multi-party collaborators. In the case of a non-restrictive authorization policy the task creator gets their task served. If the policy rule is restrictive, the policy negotiation component takes over. It first takes input from ATA, so that to define the creator's interest behind accessing the asset, and then modifies the policy rule accordingly passing it to the asset owner for confirmation. Given the asset owner's decision the task is then either satisfied or unsatisfied.

4.3 IBN Enabled PBMS

The policy negotiation framework can be integrated into a PBMS as a plug-in, enabling negotiation in policy enforcement process. A PBMS, as defined by standards organizations such as IETF and DMTF consists of four basic components as shown in Figure 3: a) the policy management tool, b) the policy repository, c) the policy enforcement point, and d) the policy decision point [21]. The policy management tool is the entry point through which policy makers interface (write, update and delete) with policies to be enforced on the system. The policy repository is a specific data store where the policies generated by the management tool are held (step A1). The PEP is the logical component that can take actions on enforcing the policies' decisions, while the PDP is the logical entity that makes policy decisions for itself or for other system elements that request such decisions. Triggered by an event that needs policy's evaluation the PEP contacts PDP (step A2), which is responsible for fetching the necessary policy from policy repository (step A3, A4), evaluates it and decides the actions that need to be enforced on PEP (step A5).

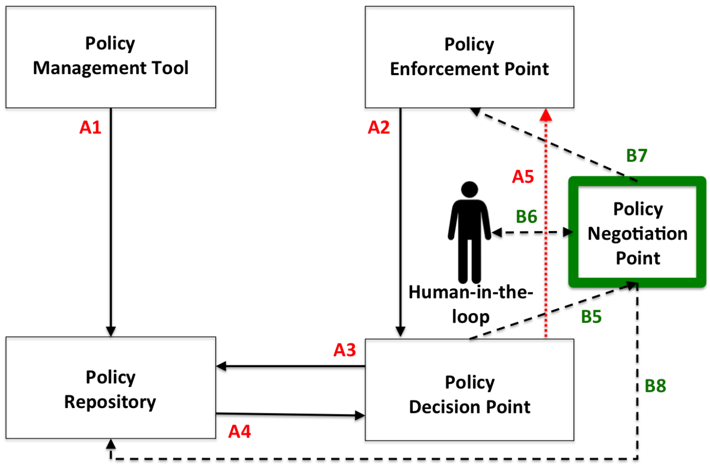

Fig. 3. IBN extended PBMS

In addition to the four basic PBMS elements, Figure 3 also includes a human-in-the-loop element, representing the roles played by the asset requestor and owner in the negotiation process. The additional component where the IBN framework resides is called Policy Negotiation Point (PNP) and lies between the PEP and PDP, interfacing also with the human-in-the-loop element. As mentioned before, the PNP is triggered to attempt to refine authorization policies when a user creates a task that cannot be served due to restrictive policies. The dashed lines show optional communication between the PBMS components, which is only established when a policy negotiation incident occurs. The red numbered part of the figure (flow paths which are prefixed by A's) describe the typical PBMS operational flow, while the green part (flow paths which are prefixed by B's) replace step A5 (red, dotted line) with the policy negotiation extension. Note that the separation between the components can be only logical when they reside in the same physical device. When PNP detects a restrictive policy (step B5) it refines it following the steps described in the following section and passes it to the asset owner for confirmation (step B6). If the asset owner accepts the refined policy rule, it is then pushed to PEP for enforcement (step B7) and either is stored in Policy Repository permanently (step B8) or can be only enforced once and then be discarded. This is on asset owner's jurisdiction. Otherwise step A5 is executed as before.

5 Achieving IBN through Policy Refinement

The negotiators in our scenario are essentially decision makers who generally lack negotiation expertise. Thus, the IBN mechanism tries to take, as much as possible, the negotiation weight off their shoulders rather than providing them the means for making proposals and trade options themselves. However, it does not exclude them completely from the negotiation process as in fully automated models. To achieve such behavior it simply applies the IBN principles described in Chefs-Orange scenario of Section 3.1, exploiting the domain model's semantics, the semantics of the polices and the seamless relation between them as they both share the same CE representation.

The objective of the negotiation is the restrictive policies. Asking the why question as in Chefs-Orange scenario, to the asset requestor side, the PNP gets as a reply the reason why they need the asset for (i.e., to get their task served). Asking the why question to the asset owners/policy authors side, it gets the reasons why they do not want to grant access to their assets respectively. The prerequisite for the PNP operation here, is to have full and accurate knowledge of the managed system. This is achieved by having unlimited and unconditional access to both domain model and policy rules of Policy Repository. Unlike the majority of the proposed PBN approaches, the human-in-the-loop negotiators in our case are ignorant of the preferences of their opponents, while their knowledge in terms of the domain model reaches only the ground of their own expertise and ownership.

Utilizing CE as the formal representation for describing the system to be managed, and the representation for expressing authorization policies, eases the human-machine communication (i.e., communication between PNP and non-IT expert negotiators) for exchanging information regarding the negotiation process in a transparent way. The human-machine communication through CE conversational agents has been described in [22] where a human-machine, machine-machine and machine-human communication protocol was presented for providing intelligence to decision makers through fusing human input, unstructured with structured information. However, trying to automate as much as possible the negotiation process, the why question is rather rhetorical here (i.e., PNP does not require input from user). In the asset requestor's case, the answer to the why question is quite simple and straightforward and the PNP is aware of it just by taking input from the Asset-Task Assignment component of Figure 2. The asset requestor clearly wants to access the asset in order to get their task served. Hence, a desired negotiation outcome as far as the asset requestor is concerned, is the derivation of a refined policy that has them included in the set of *Subject* policy block, with a positive access (i.e., canAccess) *Action*, to a *Target* set that includes the prohibited asset capable of serving their task's needs.

Inferring the answer to the why question from the asset owner's side, for understanding their interests and broadening the negotiation space, is a more challenging task. In general any application of authorization systems, aims to specify access rights to resources. A simple answer would be including the reasons why asset owners want to decline access rights to their resources in a negative authorization policy, or alternatively the reasons for granting access to their resources in a positive one. Thus, the why question from the asset owner/policy maker side can be extracted as the rationale of a policy rule. Combining the definitions from [23, 24] the rationale is the reasoning pathway from contextual facts, assumptions and decisions, through the reasoning steps, which describes the development of an artifact including details of why it was designed.

Looking carefully at a policy rule, its rationale is basically described from the policy's *Condition(s)* block. The policy R1 of Section 4.1 is rather a simple one referring deliberately to a simple scenario and this might not be easily inferred. Considering other more complex policy rules with several conditions describing for instance constraints such as the age of the requestor or their expertise this is easier inferred.

However, this is not exactly the answer to the why question we are looking for here. Considering the policies as the means for guiding systems' actions towards behaviors to achieve optimal outcomes, the *Condition(s)* policy block refers to the actions level of the policy. Our focus here is on the higher level, this of the system's behavior. Focusing on a higher level, gives us the agility to find different policies as far as the actions are concerned, that provides the same functionality in terms of behavior; and the different policies we are looking for are those which serve the needs of the asset requestors as well. Achieving this goal, leads the negotiation to a win-win outcome like the one described in Chefs-Orange scenario. The next steps describe the process for reaching such an outcome.

In the event of a restricting authorization policy, that prohibits a task creator/asset requestor to access desired resources, in order to get their task served, the PNP, is activated taking input from ATA component as shown in input step of Figure 4 (upper left hand side). The input refers to both: a) the task's needs and b) the very specific resources needed for its implementation. In a policy-based, access control system, the resources required for a task's implementation are represented by the *Target* policy rule block. If the *Target* block of the currently applied/restricting policy rule refers to a superset of the resources passed as input to PNP from ATA then the policy refinement mechanism develops as follows:

Step 1: The simplistic domain model of Figure 1 presents only the concepts involved in the smartphone scenario of Section 3.2 and their relationships. It hides however their properties. Assume that the concept Asset has a property named *Provided capability* and that the Asset instance named SMD has the Provided capability property named *Tethering.* Thus, the policy R1 by denying access to SMD, it denies access to any of SMD's provided capability or in other words denies access to any of SMD's subsets capable of serving a desired task. Thus, the IBN mechanism, trying to broaden the negotiation space, separates the SMD from its Provided capability property as shown in Step1 of Figure 4 allowing SMD's capabilities to be subject of a policy rule *Target* block.

Step 2: The concept Task has a property named *Required capability* and the Task instance *Email submission* has a number of required capabilities including that of *Tethering.* The second step of IBN process, considering input from ATA

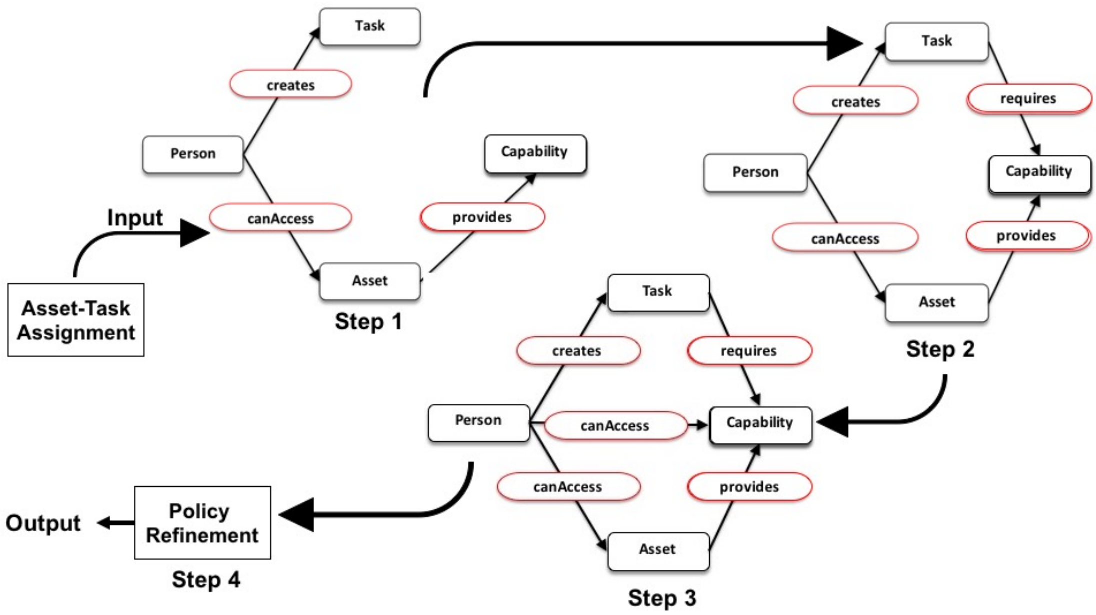

Fig. 4. IBN through Policy Refinement: Graphical representation.

regarding task's needs, it separates it from its Required capability property as shown in Step2 of Figure 4.

Step 3: In Step 3, as shown in Figure 4, now that required/provided Capability exists as standalone concept in the conceptual model, the IBN mechanism allows for a relationship and thus for an *Action* policy rule block to emerge between asset requestor (presented as Person concept) and asset's capabilities (presented as Capability concept). In other words the asset requestor (i.e., P2) now can access a subset/subsystem of the asset that provides required capabilities (i.e., *Tethering*), for their task's to be served and this can be expressed in CE policy rule and reasoned on CE conceptual model.

Step 4: This step performs the *policy refinement*[5] . The asset requestor (i.e., P2) is represented by the *Subject* block of the refined policy rule, which has as *Action* block a positive authorization action (i.e., canAccess) and its *Target* block refers to the provided by the prohibited Asset and required by the desired Task Capability (i.e., Tethering). The CE refined policy *R1-Refined* below is passed then to the asset owner for approval.

[5] Note that the term policy refinement herein refers to a different process than the policy refinement in [25], which describes the process of interpreting more general, business layer policies to more specific, system layer ones.

```
Policy R1-Refined
if
  ( there is an asset A named SMD ) and
  ( there is a Capability C named Tethering ) and
  ( there is a Person P named P2 )
then
  ( the Person P canAccess the Capability C ).
```

The asset owner P1 is in charge of approving or not the replacement of policy rule R1 from the proposed by IBN policy rule R1-Refined. In the case of approval the refined rule R1-Refined is enforced over R1 providing access to SMD's *Tethering* capability. It is then either stored in Policy Repository or can be only enforced once and then be discarded. This is on asset owner's jurisdiction.

6 Policy IBN Evaluation

Ideally, the evaluation of the behavior and effectiveness of IBN as a negotiation mechanism for policy-regulated asset sharing, should had human participants involved, operating in a collaborative environment, such as an opportunistic network scenario. Considering their own sharing constraints, they would be responsible for the approval or rejection of the refined policies proposed by IBN mechanism.

6.1 Simulation Setup

We conduct an experiment by simulating this environment, to test the policy IBN mechanism's behavior prior to carrying out a more resource-costly human participant experiment. The simulation describes an asset sharing scenario of a small, short lived opportunistic network. In the scenario, there are three basic concepts: a) the human users, b) their assets and c) the tasks they create. The users are the asset owners and in charge of sharing them with others through a PBMS. Being eager for consuming information, users create tasks that require specific resources provided by the deployed assets in order to be served. Often, task creators cannot serve their tasks just by utilizing their own resources and ask for support by their peers in the opportunistic network.

The simulated, opportunistic network scenario assumes 8 users, each one of them owns one asset. There are three types of assets, as many as the types of the tasks the users can create. Each asset type has the capability to serve a particular task type meeting its information requirements. As far as the asset sharing is concerned, it is managed through policies written by asset owners. Each user opts whether to exclusively use their assets (following concerns regarding security, privacy, performance and other) or sharing them with the others. This intention is expressed through authorization policies. We do not assume any spatial constraints in the simulated network, which implies that all the users, operate in distance where their devices have enough transmission/reception capacity to communicate with each other. Moreover, one out of three asset types has a monolithic architectural design making it capable of serving only one particular task, unlike the other two, more capable devices that can operate as platforms that provide several capabilities, able to serve more than one task types.

To better visualize the simulated scenario, think of the following vignette. Eight individuals (i.e., opportunistic network's users) go hiking across a mountain. The three types of assets are: a) a smartphone device such as the SMD described in previous sections, b) a music player equipped with transmitter/receiver and communication protocol capabilities able to communicate with other assets of the same type and c) a monolithic wearable pedometer device. The three possible tasks created by users are: a) the submission of an email, which requires internet connection provided by a smartphone device, b) a music sharing task, which is served by portable music players capable of exchanging songs and playlists with other devices of the same type and c) a step counting task, served by the monolithic pedometer device.

The IBN mechanism, following the steps described in the previous section is only capable to be applied on polylithic assets (i.e., SMD and music player device) when strict authorization policies are applied. For the implementation of the step counting task, the user needs to physically access a pedometer device. Hence, if the user that creates and need to serve a step counting task, either does not own a pedometer device or any of the pedometer devices of the network are not shareable due to strict authorization policies, the IBN mechanism is unable to provide any policy refinement.

The total number of created tasks is 100. They are created randomly by the eight users, which implies uneven number of tasks for each user. Task types are also randomly picked as do the types of the user owned assets. As it was mentioned before, the main objective of the IBN mechanism is to increase the social welfare. To measure the effect of IBN on social welfare in our scenario we use as metrics the proportion of served and dropped tasks. A task is considered dropped (i.e., unsupported by opportunistic network's deployed assets), if there are no available resources to satisfy the task utility demand.

To have a better picture of IBN effect on social welfare, we experiment with three different asset sharing models. The first and strictest one deals with very conservative (in terms of sharing policies) users where they do not share any of their devices with their peers. In this case the asset sharing is set to 0% and the user created tasks can only be served, if and only if their creators' devices are capable to do so. In the second experiment we set the asset sharing to 25% namely 25% of the total devices are shareable and finally the last and most liberal case deals with 50% asset sharing. In all three experiments we measure the proportion of served and dropped tasks when: a) the IBN mechanism is deactivated **IBN OFF** and b) the IBN mechanism is activated **IBN ON**. For all six experimental cases

- *Asset sharing 0%:* 1) IBN OFF, 2) IBN ON
- *Asset sharing 25%:* 1) IBN OFF, 2) IBN ON
- *Asset sharing 50%:* 1) IBN OFF, 2) IBN ON

we execute each simulation instance 100 times, averaging the measurements (i.e., the percentage of served and dropped tasks).

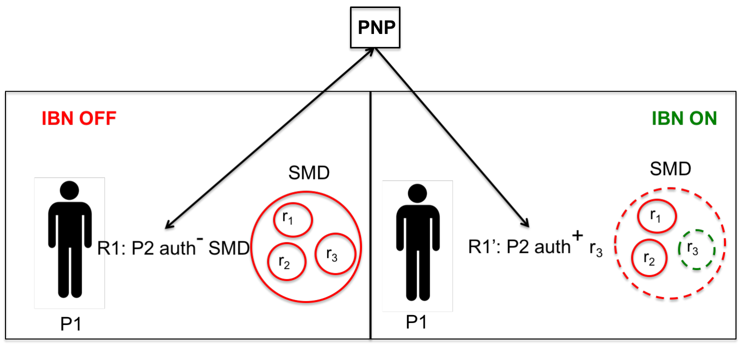

Fig. 5. PNP: IBN OFF vs IBN ON

6.2 Simulation RedLines

For simulating the approval or rejection of the refined policies (i.e., the relaxed policies provided by IBN) we utilize a mechanism called *RedLine*. We borrowed the term RedLine from the worldwide used phrase "Red line", or "cross the red line", which means a figurative point of no return or a limit past which safety can no longer be guaranteed. Each user at the beginning of the simulation, randomly gets their RedLines settled, which defines their intention to approve or reject the refined policies. Given the complexity, in terms of assets capabilities and tasks requirements, we assume, it is difficult for opportunistic network's users to write fine-grained policies to define access control on every possible combination of them. In SMD case for instance, the asset owner P1, being unable to cope with the complexity of matching SMD's capabilities and Email submission task's requirements he opts not to share any of SMD's capabilities with P2 following personal concerns. Hence, he simply expresses his constraints at higher level, setting access control policies at the assets level only. The RedLine mechanism, defines the distance between the asset owners' high-level authorization policies and their "real" willingness to share their assets or subsets of them with their peers.

This assumption is depicted in Figure 5. On the left hand side where IBN is inactive, user P1 expresses his strict constraints, in terms of SMD sharing with P2 through high-level policy rule R1, according to which the requestor user P2 is not allowed to access it and as a consequence he forbids access to any of SMD's subsystems (namely resources r_1, r_2 & r_3 and their provided capabilities). The PNP, activating IBN mechanism as shown on the right hand side of Figure 5 proposes a finer-grained rule R1' according to which the requestor user R2 can access (as the dotted line circles indicate) the necessary SMD subsystem (namely resource r_3 and its provided capability) in order to get his task served. The intention of P1 in terms of approving or rejecting the refined R1' is simulated by users' RedLines mechanism.

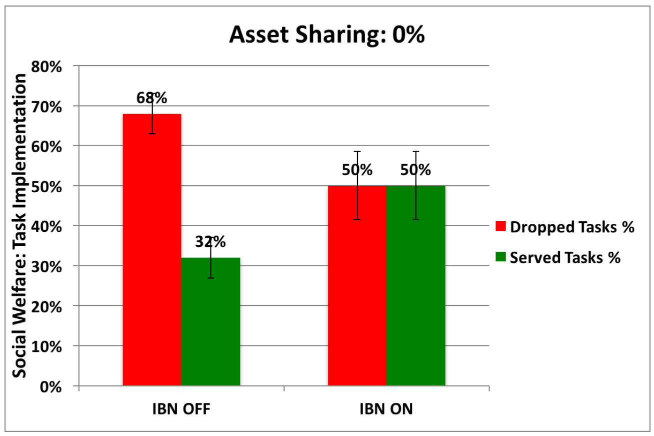

Fig. 6. IBN effect on social welfare: Asset Sharing 0%

IBN mechanism, as mentioned before, attempts to lower barriers, through policy refinement, in order to establish better collaboration through asset sharing (i.e., increase the overall number of served tasks and thus increase the social welfare) while maintaining the level of compromise from the asset owners point of view. Those users whose RedLines are more relaxed compared to their policies, represent those who believe that they do not compromise any of their concerns expressed through their initial, strict sharing policies and proceed with accepting the refined ones.

6.3 Simulation Results

The experimental results are presented through Figures 6, 7 & 8 and the error bars on clustered column charts represent (+ / -) 1 Standard Deviation. In all three sets of experiments, when the IBN mechanism is activated the social welfare in terms of task implementation is higher as expected. As it is shown the effectiveness of IBN is higher in strict environments and decreases as moving to more liberal ones. In Figure 6, for 0% asset sharing model, the IBN OFF columns indicate that only 32% of the total tasks are served meaning that given the users' information need only the one third of it can be served by their own resources. For the same sharing model, when IBN is activated the proportion of served tasks increases to 50%.

In first and most strict case Figure 6, the margin between dropped and served tasks when IBN is inactive is 36 percentage units. In IBN ON case the proportion

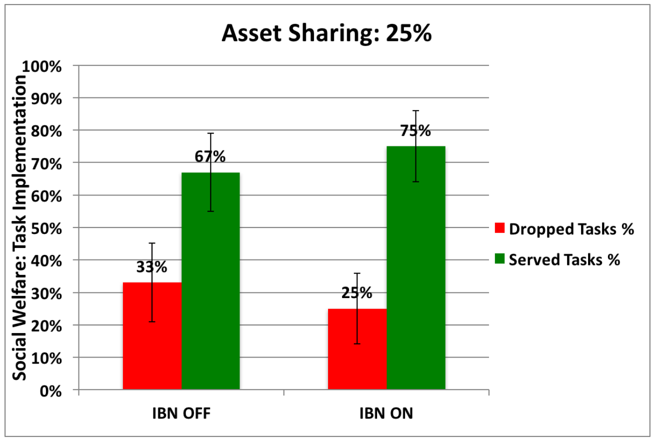

Fig. 7. IBN effect on social welfare: Asset Sharing 25%

of dropped and served tasks is even. In the second experiment and 25% asset sharing model, as shown in Figure 7, the served tasks proportion outperforms the dropped tasks' whether the IBN is active or not. With IBN ON however, the proportion of served tasks is 8 percentage units more compared to when IBN is OFF.

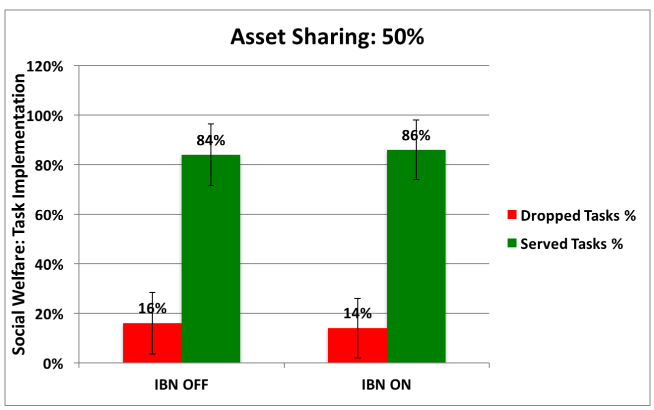

Fig. 8. IBN effect on social welfare: Asset Sharing 50%

Same trend in the last and most liberal case where half of the assets in the opportunistic network are shared with the network's users. The margin here between IBN ON and IBN OFF cases almost disappears with IBN ON case performing slightly better with 2 percentage units. Finally, as shown in Figure 8 in the simulated opportunistic network environment, if the asset sharing ratio is higher than 50% the IBN mechanism has not much to offer, while it is a useful tool in promoting collaboration through asset sharing, for stricter (in terms of asset sharing) environments.

7 Conclusion & Future Work

In summary the proposed IBN mechanism provides a policy refinement tool for revising asset sharing policies in dynamic, multi-party environments. The IBN paradigm is a good fit in multi-party environments, where collaboration is promoted, to achieve mutually satisfactory negotiation outcomes. The proposed mechanism is seamlessly interfaced with standardized PBMS and provides the means to directly negotiate with policies. Finally, the experimental evaluation indicates that when the IBN mechanism is activated the social welfare, in terms of task implementation, increases, while the effectiveness of IBN is higher in stricter environments and decreases as moving to more liberal ones.

As for the future research, there are plans for a) extending the IBN steps with regards to broadening the negotiation space considering heuristics related to users and their characteristics, such as, their team affiliation that can improve IBN's effectiveness through sharing assets and provided service horizontally (e.g. inner-team) unlike the current vertical (i.e., user-to-user) approach and b) conduct experiments involving human participants.

Acknowledgment: This research was sponsored by the US Army Research Laboratory and the UK Ministry of Defense and was accomplished under Agreement Number W911NF-06-3-0001. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the US Army Research Laboratory, the US Government, the UK Ministry of Defense or the UK Government. The US and UK Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

References

1. N. Jennings *et al.*, "Automated negotiation: Prospects, methods and challenges," *Group Decision and Negotiation*, vol. 10, no. 2, pp. 199–215, 2001.
2. D. Walton and E. Krabbe, "Commitment in dialogue: Basic concepts of interpersonal reasoning." *Albany, New York: SUNY Press*, 1995.
3. R. Fisher *et al.*, *Getting to yes: Negotiating an agreement without giving in; 2nd repr. ed.* London: Random House Business Books, 1999.
4. P. Pasquier *et al.*, "An empirical study of interest-based negotiation," *Autonomous Agents and Multi-Agent Systems*, vol. 22, no. 2, pp. 249–288, 2011.
5. M. Delaney, A. Foroughi, and W. Perkins, "An empirical study of the efficacy of a computerized negotiation support system (NSS)," *Decision Support Systems*, vol. 20, no. 3, pp. 185–197, 1997.
6. M. Wooldridge and N. R. Jennings, "Intelligent agents: theory and practice," *The Knowledge Engineering Review*, vol. 10, pp. 115–152, 6 1995.

7. P. Faratin, C. Sierra, and N. R. Jennings, "Negotiation decision functions for autonomous agents," *Robotics and Autonomous Systems*, vol. 24, no. 3 - 4, pp. 159 – 182, 1998, multi-Agent Rationality.
8. O. Martin and R. Ariel, "A course in game theory," *The MIT Press*, 1994.
9. S. Kraus, *Strategic Negotiation in Multiagent Environments.* Cambridge, MA, USA: MIT Press, 2001.
10. I. Rahwan *et al.*, "Argumentation-based negotiation," *Knowledge Engineering Review*, vol. 18, no. 4, pp. 343–375, 2003.
11. K. Seamons *et al.*, "Requirements for policy languages for trust negotiation," *3rd International Workshop on Policies for Distributed Systems and Networks, POLICY 2002*, pp. 68–79, 2002.
12. I. Jang, W. Shi, and H. Yoo, "Policy negotiation system architecture for privacy protection," *Proceedings - 4th International Conference on Networked Computing and Advanced Information Management, NCM 2008*, vol. 2, pp. 592–597, 2008.
13. M. Comuzzi and C. Francalanci, "Agent-based negotiation in cooperative processes: Automatic support to underwriting insurance policies," *ACM International Conference Proceeding Series*, vol. 60, pp. 121–129, 2004.
14. J.-F. Tang and X.-L. Xu, "An adaptive model of service composition based on policy driven and multi-agent negotiation," *Proceedings of the 2006 International Conference on Machine Learning and Cybernetics*, vol. 2006, pp. 113–118, 2006.
15. V. Cheng, P. Hung, and D. Chiu, "Enabling web services policy negotiation with privacy preserved using XACML," *Proceedings of the Annual Hawaii International Conference on System Sciences*, 2007.
16. T. Sandholm, "Distributed rational decision making," *Multiagent Systems (ed. G. Weiss) MIT Press*, pp. 201–258, 1999.
17. C. Loebecke, P. C. Van Fenema, and P. Powell, "Co-opetition and knowledge transfer," *SIGMIS Database*, vol. 30, no. 2, pp. 14–25, 1999.
18. A. Brandenburger and A. Nalebuff, "Co-opetition," *New York: Doubleday*, 1997.
19. C. Parizas, D. Pizzocaro, A. Preece, and P. Zerfos, "Managing ISR sharing policies at the network edge using controlled english," *In proceedings of Ground/Air Multisensor Interoperability, Integration, and Networking for Persistent ISR IV*, 2013.
20. D. Mott, "Summary of controlled english," *ITACS*, 2010.
21. R. Yavatkar, K. Pendarakis, and R. Guerin. (2000) A framework for policy-based admission control. http://www.ietf.org/rfc/rfc2753.txt. Accessed: 29/09/2015.
22. A. Preece, D. Braines, D. Pizzocaro, and C. Parizas, "Human-machine conversations to support mission-oriented information provision," *Proceedings of the Annual International Conference on Mobile Computing and Networking, MOBICOM*, pp. 43–49, 2013.
23. T. P. Moran, *Design Rationale: Concepts, Techniques, and Use*, J. M. Carroll, Ed. Hillsdale, NJ, USA: L. Erlbaum Associates Inc., 1996.
24. M. Dave and a. et, "Hybrid rationale and controlled natural language for shared understanding." *The 6th Annual Conference of Knowledge Systems for Coalition Operations.*, 2010.
25. A. Bandara, E. Lupu, J. Moffett, and A. Russo, "A goal-based approach to policy refinement," *Proceedings - Fifth IEEE International Workshop on Policies for Distributed Systems and Networks, POLICY 2004*, pp. 229–239, 2004.