NORMATIVE CONFLICTS
Patterns, Detection and Resolution

Georgios K. Giannikis and Aspassia Daskalopulu
Department of Computer and Communications Engineering
University of Thessaly, Gklavani 37, 38221 Volos, Greece

Keywords: Conflict Patterns, Conflict Detection, Conflict Resolution, e-Contracts, Default Logic.

Abstract: The analysis, representation and management of normative conflicts have been the focus of much research in recent years in commercial and business applications. In this paper we are concerned with normative conflicts that arise for agents engaging in electronic contracting. First, we identify a set of primitive conflict patterns and present some patterns that have not been identified in other proposals. Secondly, we use a representation of e-contracts as Default Theories, which afford us both detection and resolution of such conflict patterns.

1 INTRODUCTION

In this work we are concerned with normative conflicts that arise for agents engaging in electronic contracting, within an electronic marketplace, and we investigate an alternative representation in which we use Reiter's Default Logic (DfL) (Reiter,1980). In (Giannikis & Daskalopulu, 2006, Giannikis & Daskalopulu, 2007b) we proposed the representation of contractual norms as default rules, which are constructed dynamically from temporal representations. The resulting default theories afford us both temporal defeasible reasoning and conflict management (Giannikis & Daskalopulu, 2006). Here, we identify a set of primitive patterns for normative conflicts and show how the conflicts identified by other researchers may be seen as instances of these primitives. We also identify some patterns of normative conflict that have not been identified in other proposals. Furthermore, we discuss the way the representation of contractual norms as default rules facilitates conflict detection and resolution.

2 CONFLICT DETECTION

For the purposes of illustration consider an electronic marketplace, populated by software agents that establish and perform e-contracts on behalf of some real world parties. Let the set *Agents*={*Agent*1, *Agent*2, *Agent*3,.....} denote distinct identifiers for the various agents, and the set *Roles*={*RA*, *WA*, *MA*, *CA*, ...} denote distinct roles that agents may assume in the e-market (where *RA*, *WA*, *MA*, *CA* denote *retailer*, *wholesaler*, *mediator* and *carrier* respectively).

Consider a two-party business transaction. *Agent*1 that acts as a retailer orders some goods from the wholesaler *Agent*3. The terms of the agreement between these two agents are: *Agent*3 should see to it that the goods be delivered to *Agent*1 within 10 days from commencement (e.g., the date that the order takes place). *Agent*1, in turn, should see to it that payment be made within 21 days from the date it receives the goods. If *Agent*3 does not deliver on time, then a fixed amount is to be deducted from the original price of the goods for each day of delay and it should see to it that delivery be made by a new deadline. If *Agent*1 does not perform payment on time, then a fixed amount is to be added to the original price of the goods for each day of delay and it should see to it that payment be made by a new deadline.

Following (Daskalopulu, 2000), we may take an informal, process view of the business transaction that is regulated by the agreement. Each state offers a (possibly partial) description of the factual and normative propositions that hold true. A transition between states corresponds to an event that takes place, i.e. an action that one of the parties performs

Giannikis G. and Daskalopulu A.
NORMATIVE CONFLICTS - Patterns, Detection and Resolution.
DOI: 10.5220/0001835505270532
In *Proceedings of the Fifth International Conference on Web Information Systems and Technologies (WEBIST 2009)*, page
ISBN: 978-989-8111-81-4

or omits to perform. Normative propositions of the form:

$$NN(agent1, role1, action, agent2, role2)$$

express that *agent*1 that acts as *role*1 is in legal relation *NN* towards *agent*2 that acts as *role*2 to perform *action*, where *NN* may be *Obligation*, *Prohibition*, *Permission* and legal *Power*.

We use Reiter's Default Logic (Reiter, 1980) to represent the norms of an agreement as default rules. A default rule has the form $P:J1,J2,\ldots Jn/C$, where P is the *prerequisite*, $J=\{J1,J2,\ldots Jn\}$ is a set of *justifications* and C is the derived *consequent*. If J coincides with C, the default rule is called *normal*. The semantics of a default rule is: If P holds and the assumption J is consistent with our current knowledge, then C may be inferred.

For instance, the following default rule expresses that if an order from *Agent*1 (acting as a retailer) towards *Agent*3 (acting as a wholesaler) holds, and it is consistent to assume that *Agent*1 will become a regular client, then we may infer that *Agent*3 is legally obliged towards *Agent*1 to perform delivery:

$$\frac{Order(Agent1, RA, Agent3, WA) \;:\; BecomeRegularClient(Agent1)}{Obligation(Agent3, WA, Delivery, Agent1, RA)}$$

A Default Theory (DfT) is a pair of the form (W, D), where W is a set of closed formulae that hold, and D is a set of defaults. Rules may be used to compute *extensions* E of the default theory. A rule is applicable to a set of formulae $W \subseteq E$ if and only if $P \in E$ and $\neg J_1,\ldots, \neg J_n \notin E$. We consider grounded DfTs and we derive extensions in the manner presented in (**Error! Reference source not found.**), i.e. by maintaining consistent sets of formulae. This derivation may be conducted in stepwise manner. Thus, an agent that engages in a transaction governed by some agreement, essentially reasons with a default theory. At each time point during the business transaction the agent attempts to compute the extensions of its current DfT. Note that whenever multiple extensions are computed for a Default Theory these represent possible world views. Depending on its chosen action an agent is committed to a particular extension. The DfT contract representation allows us to detect normative conflicts by examining extensions. A normative conflict may be detected either between multiple extensions or between some extension and the current knowledge (W) of the agent. Where a conflict is detected between multiple extensions, the latter represent alternative futures for the agent; let us call these *inter-extension* conflicts. Where a conflict is detected between an extension and the current knowledge of the agent, it represents a state in which some normative violation will eventually occur; let us call these *intra-extension* conflicts. The role of conflict detection is, thus, to assist an agent to choose a course of action so that normative violations may be predicted and avoided.

The first step of conflict management involves the detection of conflicts. To this end, in section 2.1, we identify primitive patterns of normative conflict that may be spotted during the derivation of extensions of the default theory representation of an agreement. In section 2.2 we identify additional cases of normative conflicts which are not discussed already in the existing literature.

2.1 Primitive Patterns of Normative Conflicts

In what follows we use *Obligation*, *Permission*, *Prohibition* and *Power* as predicates that express normative relations between agents. We do not employ the axiomatization of any particular system of Deontic logic; specifically, we do not employ the axiomatization of Standard Deontic Logic (SDL), in which these notions are modeled as operators that are inter-defined. This is because in Standard Deontic Logic (and any system where the D scheme $\neg O\bot$, where O denotes obligation, is valid) it is not possible for an agent to bear conflicting obligations because of the D scheme. Yet, in most realistic situations, indeed in our everyday life, agents do find themselves in normative conflict. Moreover, if we were to employ SDL, permission, obligation and prohibition would be interdefined, and so all of the patterns we present in this section (section 2.1) would be reduced to three of all six patterns (Giannikis & Daskalopulu, 2007a); thus the representation would be less distinguishing.

A. Conflict between a normative notion (*NN*) and its negation. The general pattern is:

$$NN(agent1, role1, action, agent2, role2)$$

$$\neg NN(agent1, role1, action, agent2, role2)$$

This is the common syntactical conflict that arises when an agent has contradictory knowledge. All other approaches, without any exception, refer to this type of conflict. In policy-based approaches, when the normative notion is *Obligation* it is called *positive-negative conflict of modalities* (Moffett & Sloman, 1993). This type of conflict never actually arises in our representation, where

norms are represented as defaults, because the derivation of extensions preserves consistency. It may, however, be identified as a potential conflict, when multiple extensions are computed.

B. Conflict between the prohibition to perform an action and the simultaneous permission or obligation to perform the same action. Once again, all previous research approaches refer to this type of conflict. In (Moffett & Sloman, 1993) and (Lupu & Sloman, 1999) these conflicts are called *conflicts between authority policies* (sub-pattern B1: Prohibition vs Permission) and *conflict between authority and imperatival policies* (sub-pattern B2: Prohibition vs Obligation) respectively.
Consider, for instance, the following default theory (W, D) where:
W={*Order*(*Agent*1, *RA*, *Agent*3, *WA*)}
and D={

$$\frac{Order(Agent1, RA, Agent3, WA) : WellKnownDebtor(Agent1)}{Prohibition(Agent3, WA, Delivery, Agent1, RA)}$$

$$\frac{Order(Agent1, RA, Agent3, WA) : Permission(Agent3, WA, Delivery, Agent1, RA)}{Permission(Agent3, WA, Delivery, Agent1, RA)}$$ }

The first default denotes that if an order from *Agent*1 (acting as retailer) towards *Agent*3 (acting as wholesaler) holds, and it is consistent to assume that *Agent*1 is related to a well known debtor then we may infer that *Agent*3 is prohibited to perform delivery. Similarly, the second default expresses that if an order from *Agent*1 towards *Agent*3 holds, and it is consistent to assume that *Agent*3 is permitted to perform delivery, then we may infer that *Agent*3 is permitted to perform delivery towards *Agent*1. *Agent*3 may find itself in a conflicting state after applying the two defaults sequentially. We denote this type of conflict as B1. Note that special terms, such as *WellKnownDebtor(agent)*, *BecomeRegularClient(agent)* or *IsRegularClient(agent)* among others, are used only for the purposes of illustration and are not binding to the characterization of domain-independent conflict patterns.
In the same spirit, let us replace the second default shown above with the following:

$$\frac{Order(Agent1, RA, Agent3, WA) : Obligation(Agent3, WA, Delivery, Agent1, RA)}{Obligation(Agent3, WA, Delivery, Agent1, RA)}$$

Once again *Agent*3 is in conflict. We denote this conflict between Prohibition and Obligation as B2.

C. Conflict between an obligation to perform *action* and the simultaneous obligation or permission to perform ¬*action*. Here ¬*action* denotes a negative action, and the issue of representing negative actions has concerned researchers (e.g. (Royakkers, 1998) regards them as actions that do not lead to the successful fulfillment of a norm). We have not developed special semantics for the representation of negative actions; we merely regard such expressions as denoting either performance of some action other than the negative one, or as idleness (non performance of any action). This case arises, also, in Lee (Lee, 1988) and Abrahams (Abrahams & Bacon, 2002) who use the term *Waive*.
For example consider the following DfT where:
W={*Order*(*Agent*1, *RA*, *Agent*3, *WA*)}
and D={

$$\frac{Order(Agent1, RA, Agent3, WA) : BecomeRegularClient(Agent1)}{Obligation(Agent3, WA, Delivery, Agent1, RA)}$$

$$\frac{Order(Agent1, RA, Agent3, WA) : WellKnownDebtor(Agent1)}{Obligation(Agent3, WA, \neg Delivery, Agent1, RA)}$$ }

D. Conflict between the power to perform an action and the simultaneous prohibition to perform the same action. This type of conflict is also noted in (Abrahams & Bacon, 2002).
For instance consider the following DfT:
W={*Order*(*Agent*1, *RA*, *Agent*3, *WA*)}
and D={

$$\frac{Order(Agent1, RA, Agent3, WA) : Power(Agent3, WA, Delivery, Agent1, RA)}{Power(Agent3, WA, Delivery, Agent1, RA)}$$

$$\frac{Order(Agent1, RA, Agent3, WA) : WellKnownDebtor(Agent1)}{Prohibition(Agent3, WA, Delivery, Agent1, RA)}$$ }

One may argue that in this case there is no conflict and, consequently, that there is no need for conflict resolution. Indeed, legal power to perform an action goes hand-in-hand with permission to exercise it, according to formal definitions of institutional power ((Makinson, 1986, Jones & Sergot, 1996)). Hence, there is a conflict here, albeit some may perceive it as a conflict between permission and prohibition to exercise a certain power.

E. Conflict between two obligatory distinct actions, when it is impossible to do both at the same time. This corresponds to Horty's moral dilemma (Horty, 1994).
For instance consider the following DfT where:
W={*Order*(*Agent*1, *RA*, *Agent*3, *WA*),
Order(*Agent*2, *RA*, *Agent*3, *WA*),
no simultaneous performance of actions is possible}
and *D*={

$$\frac{Order(Agent1, RA, Agent3, WA) \;:\; BecomeRegularClient(Agent1)}{Obligation(Agent3, WA, Delivery1, Agent1, RA)}$$

$$\frac{Order(Agent2, RA, Agent3, WA, T1) \;:\; IsRegularClient(Agent1)}{Obligation(Agent3, WA, Delivery2, Agent2, RA)}$$ }

*Agent*3 bears two obligations that cannot be simultaneously satisfied.

F. Conflict between an obligation and the negation of the agent's permission or power to perform it. The negation of an agent's permission/power to perform an action may be explicitly derived from the agent's knowledge base (sub-pattern F1) or it may be derived from a possibly incomplete knowledge base, through the absence of explicit information (sub-pattern F2). For instance consider the following default theory where:
W={*Order*(*Agent*1, *RA*, *Agent*3, *WA*)}
and *D*={

$$\frac{Order(Agent1, RA, Agent3, WA) \;:\; BecomeRegularClient(Agent1)}{Obligation(Agent3, WA, Delivery, Agent1, RA)}$$

$$\frac{Order(Agent1, RA, Agent3, WA) \;:\; WellKnownDebtor(Agent1)}{\neg Permission(Agent3, WA, Delivery, Agent1, RA)}$$ }

Now consider a DfT that contains the first of the defaults above and in place of the second, the following:

$$\frac{Order(Agent1, RA, Agent3, WA) \;:\; \neg Permission(Agent3, WA, Delivery, Agent1, RA)}{\neg Permission(Agent3, WA, Delivery, Agent1, RA)}$$

If the agent's knowledge base does not contain an explicit permission, then the justification of this default will be satisfied, and hence its conclusion will be drawn.

2.2 Additional Patterns

Here are some additional cases of normative conflict that we have identified, which are not discussed already in the existing literature. We mention them separately because, although they may be reduced to the primitive patterns, there is additional information that may be exploited to facilitate conflict resolution.

Type of action-based conflicts. A common feature of e-contracts is the so called Contrary-to-Duty structures (Prakken & Sergot, 1996). An agent's contractual obligations may be divided in two types. *Prima facie* obligations that concern the performance of actions that are in principle stipulated by the agreement and secondary obligations that concern the performance of reparatory actions; the latter apply only when violations of *prima facie* obligations happen.

An agent may, thus, bear two distinct obligations (for instance of the kind described by E), where one is primary and the other is secondary. This qualification may be helpful in conflict resolution. The general pattern is:

Obligation(agent1, role1, action, agent2, role2)

Obligation(agent1, role1, reparatoryaction, agent3, role3)

Agreement-based conflicts. An agent may find itself in a conflicting state because it is engaged in multiple contracts. For instance a wholesaler may be obliged to perform two distinct deliveries to two distinct retailers as dictated by two distinct agreements. This situation may be regarded as the generalization of pattern E discussed earlier, because in this case the important information is the distinction between the contracts. The additional information that the two norms stem from two agreements, may be exploited for the purposes of conflict resolution. The general pattern is:

Obligation(contract1, agent1, role1, action1, agent2, role2)

Obligation(contract2, agent1, role1, action2, agent3, role3)

where normative propositions of the form *NN(contract, agent1, role1, action, agent2, role2)* express that according to *contract*, *agent*1 that acts as *role*1 is in legal relation *NN* towards *agent2* that acts as *role2* to perform *action*.

Note that this conflict pattern is different form the one presented in (Herrestad & Krogh, 1995). The key notion here is the different contracts an agent has to comply with. Different contracts may be established towards different agents or even towards the same agent.

Conflicts between assumptions and knowledge. A conflict may arise not only as a result of an agent's explicit knowledge but also between its knowledge and its current assumptions or even between distinct assumptions.

For example, according to the following DfT the prohibition that derives from the second default contradicts not only with obligation that derives from the first default, but also with the assumption of the first default (*Permission*):

W={*Order*(*Agent*1, *RA*, *Agent*3, *WA*)}

and *D*={

$$\frac{Order(Agent1, RA, Agent3, WA) : Permission(Agent3, WA, Delivery, Agent1, RA)}{Obligation(Agent3, WA, Delivery, Agent1, RA)}$$

$$\frac{Order(Agent1, RA, Agent3, WA) : WellKnownDebtor(Agent1)}{Prohibition(Agent3, WA, Delivery, Agent1, RA)}$$ }

3 CONFLICT RESOLUTION

Conflict resolution in DfL may be performed using Brewka's (Brewka, 1994) proposal that enables us to define and apply priorities on default rules dynamically.

Brewka defines a Preferential Default Theory (PDfT) as a triple (*W*, *D*, *name*), where name is a function that assigns names to default rules D. The extension of a PDfT is derived in the same way as in a DfT.

What makes PDfTs really useful is that the ascription of priorities to default rules may, itself, be done dynamically. Using dynamic priorities, we generate preferred extensions, each of which indicates a transaction plan. Priorities amongst ground defaults may be defined dynamically either by making different assumptions or by specifying domain-dependent criteria. The general pattern for ascribing priorities dynamically takes the form of a default rule:

$$\frac{Rule(d1, v1) \wedge Rule(d2, v2) \wedge criterion : Assumptions}{d1 < d2}$$

Here *d1*, *d2* are variables that denote names of ground defaults; *Rule(d1,v1)* denotes a ground default *d1* and its set of entities of interest *v1*. The intended interpretation of this rule is: if two defaults *d1* and *d2* apply and some criterion is satisfied between entities of interest, then *d1* takes priority over *d2*, if certain assumptions may consistently be made.

Three general strategies for defining such criteria have been discussed in the literature, namely hierarchies of entities of interest, time and specificity of norms. Thus, given a particular normative conflict, different resolution strategies may be applied depending on our specific criterion of interest.

In (Giannikis & Daskalopulu, 2006) we presented a temporal representation of normative relations in DfL, which takes into account the *external* time of a norm (i.e. the time at which it comes into force) and the *internal* time of a norm (i.e. the time stipulated for its satisfaction, its deadline). Here, a formula of the form:

NN(*agent*1, *role*1, *action*, *time*2, *agent*2, *role*2, *time*1)

denotes that at time point *time*1 *agent*1 (acting as *role*1) is in legal relation *NN* towards *agent2* (acting as *role*2) to perform *action* by *time*2. For instance, consider the case where two norms (*D*1 and *D2*) that define conflicting obligations for *Agent*3 are active. The first one is initiated at time *ET1* and it is towards retailer *Agent*1 who is a regular client. It sets an obligation to perform delivery until *IT1*. The second one is towards retailer *Agent2*, it is initiated at *ET2* and defines a reparatory obligation to perform delivery until *IT2*. The relation between time points is as follows: *ET1* < *ET2* < *IT2* < *IT1*. There is information that can be used to determine different conflict resolution criteria. The strategy of temporality based on external time may give priority to *D1* as it was initiated first. On the other hand, temporality based on internal time may give priority to *D2* since it has a shorter deadline. Another alternative, using the strategy of hierarchy is to give precedence to *D1*, because *Agent1*, as a regular client, takes precedence over *Agent2*. Or, we may give precedence to *D2*, because it concerns a reparatory action, if we choose to assign higher priority to secondary norms over primary ones. It should be clear that various combinations of these

criteria may also be defined based on the agent's current knowledge and the assumptions it makes.

4 CONCLUSIONS AND FUTURE WORK

In (Giannikis & Daskalopulu, 2007b) we proposed the representation of e-contracts as default theories that can be constructed dynamically from event calculus representations. This technique affords us the ability to perform temporal reasoning, defeasible reasoning and conflict management. In this work, we presented a set of normative conflict patterns that may be encountered in e-contracts, and recorded some conflicts that have not been identified yet in other proposals. Our current work focuses on developing a computational tool based on Reiter's DfL and its major variations, that supports temporal defeasible reasoning as well as conflict detection and resolution as presented in this paper and in (Giannikis & Daskalopulu,2006).

REFERENCES

Abrahams, Alan S., & Bacon, Jean M. 2002 (May). The Life and Times of Identified, Situated, and Conflicting Norms. *Pages 3–20 of: DEON'02: Deontic Logic in Computer Science, 6th International Workshop on Deontic Logic in Computer Science*.

Antoniou, Grigoris. 1999. A tutorial on default logics. *ACM Computer Surveys*, **31**(4), 337–359.

Brewka, Gerhard. 1994. Reasoning about priorities in default logic. *Pages 940–945 of: AAAI'94: Proceedings of the twelfth national conference on Artificial intelligence (vol. 2)*. Menlo Park, CA, USA: American Association for Artificial Intelligence.

Daskalopulu, Aspassia. 2000. Modeling Legal Contracts as Processes. *Pages 1074–1079 of: Legal Information Systems Applications, 11th International Workshop on Database and Expert Systems Applications (DEXA'00)*. IEEE Computer Society.

Giannikis, Georgios K., & Daskalopulu, Aspassia. 2006. Defeasible Reasoning with e-Contracts. *Pages 690–694 of: IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT 2006)*. Hong Kong, China: IEEE Computer Society.

Giannikis, Georgios K., & Daskalopulu, Aspassia. 2007a (February). *Normative Conflict Management with Default Logic*. Tech. rept. 5154/1, A006.3ΓΙΑ. Department of Computer and Communications Engineering, University of Thessaly.

Giannikis, Georgios K., & Daskalopulu, Aspassia. 2007b. The Representation of e-Contracts as Default Theories. *Pages 963–973 of:* Okuno, H.G., & Ali, M. (eds), *Proceedings of 19th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems (IEA/AIE 2007)*. LNAI 4570. Kyoto, Japan: Springer-Verlag Berlin Heidelberg.

Herrestad, H., & Krogh, C. 1995. Deontic Logic Relativised to Bearers and Counterparties. *Pages 453 – 522 of: Anniversary Anthology in Computers and Law, Ed. J. Bing. and O. Torrund*.

Horty, John F. 1994. Moral Dilemmas and Nonmonotonic Logic. *Journal of Philosophical Logic*, **23**(1), 35–65.

Jones, Andrew J.I., & Sergot, Marek J. 1996. A formal characterisation of institutionalised power. *Journal of the IGPL*, **4**(3), 427–443.

Lee, Ronald M. 1988. Bureaucracies as Deontic Systems. *ACM Trans. Inf. Syst.*, **6**(2), 87–108.

Lupu, Emil, & Sloman, Morris. 1999. Conflicts in Policy-Based Distributed Systems Management. *IEEE Trans. Software Eng.*, **25**(6), 852–869.

Makinson, David. 1986. On the formal representation of rights relations. *Journal of Philosophical Logic*, **15**(4), 403–425.

Moffett, Jonathan D., & Sloman, Morris S. 1993. Policy Conflict Analysis in Distributed System Management. *Journal of Organizational Computing*.

Prakken, Henry, & Sergot, Marek J. 1996. Contrary-to-Duty Obligations. *Studia Logica*, **57**(1), 91–115.

Reiter, Raymond. 1980. A Logic for Default Reasoning. *Artif. Intell.*, **13**(1-2), 81–132.

Royakkers, Lamber M. M. 1998. *Extending Deontic Logic for the Formalisation of Legal Rules*. Kluwer Academic Publishers.