A Decidable Logic for Complex Contracts

Cristian Prisacariu
joint work with Gerardo Schneider

Precise Modeling and Analysis group (PMA),
University of Oslo

21st Nordic Workshop on
Programming Theory (NWPT’09)

15th October 2009, Lyngby, Denmark.
Aim and Motivation

- We want a **formal language** for specifying/writing contracts with the required theoretical apparatus (+ tool support).
- **Analyze** contracts statically
  - Detect contradictions/inconsistencies or superfluous contract clauses
  - Analyze the contract by checking if predefined properties hold.
  - Determine the obligations (permissions, prohibitions) of a signatory
- **Monitor** contracts (dynamically) [Molina-Jimenez et al.]
  - At run-time to ensure the contract is respected
  - In case of contract violations, act accordingly
- **Tackle** the negotiation process (automatically?)
- **Develop a theory of contracts**\(^1\)
  - Contract composition, Subcontracting, Conformance between a contract and the governing policies

\(^1\)Current and Future work
Outline - (you will see in this talk)

$\mathcal{CL}$, a logic for electronic contracts + (some) theoretic apparatus.

- The $\mathcal{CL}$ logic combines:
  - Deontic Logic (modalities over actions) with
  - Propositional Dynamic Logic (of actions).

- Allows specification of abstractions of contracts in the particular style of $\mathcal{CL}$, and includes:
  - synchronous actions (i.e., actions “done at the same time”)
  - conditional obligations, permissions, and prohibitions
  - (nested) contrary-to-duty (thought like exceptions enforced after violation is detected)
  - statements about the outcome of actions in an electronic contract

- Semantics on normative structures.

- Can do (within the same syntax):
  - model checking of properties on the abstract model of the contract
  - run-time monitoring of the actions of the parties in the contract
Deontic e-Contracts

- Based on deontic logic and combined with other modal logics
- It contains constructs to specify at least the legal notions
  - Obligations, Permissions, and Prohibitions
- A deontic e-contract can be obtained
  - From a conventional contract (legal/financial context)
  - Written directly in a formal specification language (web services, components, OO)
- It allows formal reasoning
  - like: theorem proving, model-checking, runtime monitoring

Definition

A contract is a document which engages several parties in a transaction and stipulates commitments (obligations, rights, prohibitions), as well as penalties in case of contract violations.
(Standard) Deontic Logic

In One Slide

- Concerned with **normative** and **moral** notions [von Wright et al.]
  - obligation, permission, prohibition, optionality, power, immunity, etc
- Focus on
  - The logical consistency of the above notions
  - The faithful representation of their intuitive meaning in law, moral systems, or business organizations
- Difficult to avoid **puzzles** and **paradoxes**
- Approaches
  - ought-to-do: expressions consider names of actions [Meyer, Segerberg]
    - “The Internet Provider ought to send a password to the Client”
  - ought-to-be: expressions consider state of affairs (results of actions)
    - “The average bandwidth ought to be more than 20kb/s”
- Important notions:
  - conditional obligations, permissions, and prohibitions
  - contrary-to-duty and contrary-to-prohibition [Prakken et al.]
Propositional Dynamic Logic (PDL)

- Is the logic of regular programs (the logic of actions). [Pratt]
- The dynamic modality $[\beta]C$. [Fischer & Ladner, Harel, ...]
- Semantics over Kripke-like structures (regular relational structures)
- Variants:
  - over deterministic structures [Parikh, Ben-Ari & Halpern & Pnueli]
  - with intersection of actions (undecidable over det. struct.) [Harel]
  - with loop or repeat for infinite programs [Harel & Pratt, Streett]
  - with action negation (undecidable in general; decidable with negation of only atomic actions)
- embeds Hoare logic, temporal logics, many modal and epistemic logics

Example (reading the modalities $[a]C$ and $\langle a \rangle C$)

$\langle a \rangle C$ non-det. - “action $a$ can make formula $C$ true”
$\langle a \rangle C$ det. - “action $a$ will make formula $C$ true”
$[a]C$ (non-)det. - “action $a$ may always make formula $C$ true”
The Contract Specification Language $\mathcal{CL}$

\[ C := \phi_0 \mid O_\mathcal{C}(\alpha) \mid P(\alpha) \mid F_\mathcal{C}(\alpha) \mid C \rightarrow C \mid [\beta]C \mid \perp \]

\[ \alpha := 0 \mid 1 \mid a \mid \alpha \times \alpha \mid \alpha \cdot \alpha \mid \alpha + \alpha \]

\[ \beta := 0 \mid 1 \mid a \mid \beta \times \beta \mid \beta \cdot \beta \mid \beta + \beta \mid \beta^* \mid C? \]

- $\rightarrow$ and $\perp$ are the classical propositional implication and false
- $O_\mathcal{C}(\alpha), P(\alpha), F_\mathcal{C}(\alpha)$ are the deontic modalities over actions (i.e. statements about the deontic status of the actions of a contract) with the associated reparations $\mathcal{C}$ (encoding our notion of contrary-to-duty)
- $[\beta]$ action parameterized dynamic modality of PDL [Fischer&Ladner] (encoding statements about the outcome of the actions)
- $\alpha$ and $\beta$ are synchronous actions [Milner,Berry] (encoding complex actions of a contract)
- $\phi_0 \in \Phi_B$ are propositional constants [Meyer,Segerberg] (encoding basic statements about the state of the contract)
Particularities of $\mathcal{CL}$

Synchronous Actions

- Actions, $a \in A_B$ (finite set of basic actions):
  \[
  \alpha := 0 \mid 1 \mid a \mid \alpha \times \alpha \mid \alpha \cdot \alpha \mid \alpha + \alpha
  \]
  \[
  \beta := 0 \mid 1 \mid a \mid \beta \times \beta \mid \beta \cdot \beta \mid \beta + \beta \mid \beta^* \mid C?
  \]

- synchrony ($\times$), “actions done at the same time” [Milner, Berry]
- causal sequence ($\cdot$), unrestricted choice ($+$), bounded repetition ($\ast$), guarding tests ($?$) [Kozen, Harel, Pratt] [Meyer, Segerberg]

synchrony axiom: $(\alpha \times \alpha) \times (\beta \times \beta) = (\alpha \times \beta) \cdot (\alpha \times \beta) \quad \forall \alpha, \beta \in A_B^\times$

Example

- “Obliged to provide telephone and internet at the same time.”
  \[
  O(t \times i)
  \]
- “It is forbidden to drive and talk at the mobile.”
  \[
  F(d \times m)
  \]
Particularities of $\mathcal{CL}$

Conflicts, Violations, Contrary-to-duty

- **conflict relation** $\#$ on basic actions [Nielsen & Winskel, Berry]
  
  E.g.: conflicting actions (which cannot be done at the same time) like: “go west” and “go east” may give an inconsistency if $O(a) \land O(b)$

- **violation of obligations as action negation** $\overline{\alpha}$ [Broersen et al.]
  
  Intuitively $\overline{\alpha}$ is a choice of all actions which take us outside $\alpha$
  
  E.g.: consider two atomic actions $a$ and $b$ then $a \cdot b$ is $b + a \cdot a$

- **contrary-to-duty** (temporal CTDs)
  
  - Reparation enforced after violating action is detected [Meyer]
  - Not enforced in the same world as the primary obligation [Governatori & Rotolo, Prakken & Sergot, Makinson & van der Torre]
  - Not the same as $O(\alpha) \land [\overline{\alpha}]C$ [Meyer et al.]

  The reparation $C$, is attached to the obligation.
A **Normative Structure** is:  

\[ N = (\mathcal{W}, R_{2^A_B}, \mathcal{V}, \varrho) \]

- \( \mathcal{W} \) is a set of states/nodes
- \( \mathcal{V} : \Phi_B \rightarrow 2^{\mathcal{W}} \) is a valuation function of the propositional constants
- \( R_{2^A_B} : 2^{A_B} \rightarrow 2^{\mathcal{W} \times \mathcal{W}} \) is a function returning for each label a partial function (i.e. deterministic struct.) on the set of worlds
- \( \varrho : \mathcal{W} \rightarrow 2^\Psi \) is a marking function of worlds with markers from \( \Psi = \{\circ_a, \bullet_a \mid a \in A_B\} \)

**[van der Meyden, Castro & Maibaum]**

restriction: \( \forall w \in \mathcal{W}, \forall a \in A_B \text{ is not the case that } \circ_a \in \varrho(w) \) and \( \bullet_a \in \varrho(w) \)

\[ N, i \models O_C(\alpha) \text{ iff } I(\alpha) S_i N, \text{ and } \]

\[ \forall t \xrightarrow{\gamma} t' \in I(\alpha), \forall s \xrightarrow{\gamma'} s' \in N \text{ s.t. } t S s \wedge \gamma \subseteq \gamma' \]

then \( \forall a \in A_B \text{ if } a \in \gamma \text{ then } \circ_a \in \varrho(s'), \text{ and } \)

\[ \forall s \xrightarrow{\gamma'} s' \in N_{rem}^{I(\alpha); i}, \forall a \in A_B \text{ if } a \in \gamma' \text{ then } \circ_a \not\in \varrho(s'), \text{ and } \]

\[ N, s \models C \text{ } \forall s \in N \text{ with } t \hat{S} s \wedge t \in leafs(I(\alpha)). \]
Motivating Properties of $\mathcal{CL}$

Why all that complication?

**Answer:** to capture notions we find natural in legal contracts

1. **main:** $\models O_c(\alpha) \land O_c(\gamma) \rightarrow O_c(\alpha \times \gamma)$
2. **avoid conflicts:** $\models \neg (O_c(\alpha) \land F_c(\alpha))$ (the restriction on $\varrho$)  
   if $\alpha \neq \gamma$ then $\models \neg (O_c(\alpha) \land O_c(\gamma))$ (# the conflicting relation)
3. **required implications:** if $\alpha = \gamma$ then $\models O_c(\alpha) \iff O_c(\gamma)$  
   $\models O_c(\alpha) \rightarrow P(\alpha)$  
   $\models F_c(\alpha) \rightarrow F_c(\alpha \times \gamma)$  
   $\models P(\alpha \cdot \beta) \leftrightarrow P(\alpha) \land [\alpha]P(\beta)$  
   $\models F(\alpha \cdot \beta) \leftrightarrow F(\alpha) \lor \langle \alpha \rangle F(\beta)$  
4. **avoid unnatural implications:**  
   $\not\models O_c(\alpha) \rightarrow O_c(\alpha \times \gamma)$  
   $\not\models O_c(\alpha + \gamma) \rightarrow O_c(\alpha \times \gamma)$  
   $\not\models O_c(\alpha) \rightarrow O_c(\alpha + \gamma)$  
   $\not\models O_c(\alpha) \rightarrow O_c(\alpha + \gamma)$  
   $\not\models F_c(\alpha \times \gamma) \rightarrow F_c(\alpha)$  
   $\not\models P(\alpha + \gamma) \rightarrow P(\alpha)$
Theoretical results

- Completeness of the algebra of synchronous actions w.r.t. standard models of sets of guarded concurrent strings. (as corollary we can use in the semantics of $\mathcal{CL}$ the equivalent guarded concurrent trees of the actions)

- Decidability of the satisfiability problem for $\mathcal{CL}$
  (complexity of the satisfiability is at least exponential)

- Tree model property for $\mathcal{CL}$

- Runtime monitoring using a trace semantics for $\mathcal{CL}$
  (uses automata theoretic techniques)

- Trace semantics is consistent with the full semantics of $\mathcal{CL}$ based on normative structures

- Properties of $\mathcal{CL}$ in the form of validities and non-validities
  (which should lead to an axiomatization)
Conclusion

You have seen $\mathcal{CL}$:

- An action-based specification language for electronic contracts
- How the language combines
  - Propositional Dynamic Logic over synchronous actions with
  - Deontic Logic over actions.
- Semantics on normative structures
  (deterministic and with a simple marking function)
- $\mathcal{CL}$ was used for runtime monitoring
- $\mathcal{CL}$ in a restricted form was used for model checking
- $\mathcal{CL}$ has a particular view of CTDs
- $\mathcal{CL}$ combines several particular notions:
  - structured actions (regulated by an action algebra)
  - synchronous actions
  - conflicting definitions
  - action negation (to encode violations)
Current and Future Work

Theoretical tools for $\mathcal{CL}$

- Axiomatization!?
- Tableaux based proof system!?

Extensions of $\mathcal{CL}$

- deadlines
- parties (as directed obligations? or types for actions?)
- classical timeless CTDs (dyadic operators over actions?)

Controlled natural language for $\mathcal{CL}$

(ACL and DRS)

Applications of $\mathcal{CL}$ (in computer science)

- to behavior interfaces in OO
- to service-oriented architectures
Thank you!
Related Work

- C. Molina-Jimenez et al.: monitoring contracts written directly as FSM
- Fischer&Ladner, D.Harel, V.Pratt: Propositional Dynamic Logic PDL
- R.Milner, G.Berry: on SCCS and Synchrony model
- J.J. Meyer, K.Segerberg: Dynamic Deontic Logic (over actions)
- J.J. Meyer&F.Dignum&J.Wieringa et al.: combination with temporal logic, time, encoding into modal $\mu$-calculus, ...
- M.Nielsen&G.Winskel: concurrency models and conflict relation
- J.Broersen et al.: action negations and Deontic logic
- G.Governatori&A.Rotolo, H.Prakken&M.Sergot, D.Makinson&L.van der Torre: contrary-to-duty
- van der Meyden: Deontic Logic of Permissions (over actions)
- P.Castro&T.Maibaum: complete, compact deontic action logic (tableaux system)
- D.Kozen et al.: on Kleene algebra (the algebra behind actions)
- J.F.Groote et al.: mCRL2 complex action algebra
- G.H.von Wright: Deontic Logic :-)}
Contracts and Informatics

1. “Programming by contract” or “Design by contract” (e.g., Eiffel)
   ▶ Relation between pre- and post-conditions of routines, method calls, invariants, temporal dependencies, etc

2. In the context of web services (SOA)
   ▶ Service-Level Agreement, usually written in an XML-like language (e.g. WSLA)

3. Behavioral interfaces
   ▶ Specify the sequence of interactions between different participants. The allowed interactions are captured by (sets of) correct traces

4. Contractual protocols (coordination models)
   ▶ To specify the interaction (coordination) between communicating entities

5. Policy in multi-agent systems (agent societies)

6. “Deontic e-contracts”: representing Obligations, Permissions, Prohibitions, Power, etc
Predecessors

- **Deontic Logics of Actions** [von Wright, Segerberg]
  
  A foundation of actions is important for deontic logic. The first investigations in this direction; (pioneers)

- **Dynamic Deontic Logic (DDL)** [Meyer’88]
  
  The first encoding of deontic modalities in PDL (in a logic of actions)
  
  \[
  O(a) \models [\overline{a}]V \quad \text{not performing } a \text{ sees a violation marker } V
  \]

- **Dynamic Logic of Permission** [van der Meyden’90]
  
  More complex markers (for permissions this time)

- **Deontic Logic in modal }\mu\text{-calculus** [Broersen&Wieringa&Meyer]
  
  encoding in a more expressive logic than PDL. Technically cleaner. But aims at different properties than DDL.

- **Deontic Logic and Temporal or Modal Logics** [Dignum et al.]
  
  Deontic operators are stand alone but combined with temporal modalities for more expressivity.