On the Complexity of Input/Output Logic

Xin Sun[1] and Diego Agustín Ambrossio[12]

[1] Faculty of Science, Technology and Communication, University of Luxembourg, Luxembourg
xin.sun@uni.lu

[2] Interdisciplinary Centre for Security, Reliability and Trust, University of Luxembourg, Luxembourg
diego.ambrossio@uni.lu

Abstract. The complexity of input/output logic has been sparsely developed. In this paper we study the complexity of four existing input/output logics. We show that the lower bound of the complexity of the fulfillment problem of these input/output logics is coNP, while the upper bound is either coNP, or P^{NP}.

1 Introduction

In the first volume of the handbook of deontic logic and normative systems [4], input/output logic [6–9] appears as one of the new achievements in deontic logic in recent years. Input/output logic takes its origin in the study of conditional norms. Unlike the modal logic framework, which usually uses possible world semantics, input/output logic adopts mainly operational semantics: a normative system is conceived in input/output logic as a deductive machine, like a black box which produces normative statements as output, when we feed it descriptive statements as input. For a comprehensive introduction to input/output logic, see Parent and van der Torre [9]. A technical toolbox to build input/output logic can be found in Sun [12].

While the semantics and application of input/output logic has been well developed in recent years, the complexity of input/output logic has not been studied yet. In this paper we fill this gap. We show that the lower bound of the complexity for the fulfillment problem of four input/output logics is coNP, while the upper bound is either coNP or P^{NP}.

The structure of this paper is as follows: we present a summary of basic concepts and results in input/output logic and some notes in complexity theory, in Section 2. In Section 3 we study the complexity of input/output logic. We point out some directions for future work and conclude this paper in Section 4.

2 Background

2.1 Input/output logic

Makinson and van der Torre introduce input/output logic as a general framework for reasoning about the detachment of obligations, permissions and institutional facts from conditional norms. Strictly speaking input/output logic is not a single logic but a family of logics, just like modal logic is a family of logics containing systems K, KD, S4, S5, ... We refer to the family as the input/output framework. The proposed framework has been applied to domains other than normative reasoning, for example causal reasoning, argumentation, logic programming and non-monotonic logic, see Bochman [2].

Let $\mathbb{P} = \{p_0, p_1, \ldots\}$ be a countable set of propositional letters and PL be the propositional language built upon $\mathbb{P}$. Let $N \subseteq PL \times PL$ be a set of ordered pairs of formulas of PL. We call N a normative system. A pair $(a, x) \in N$, call it a *norm*, is read as "given a, it ought to be x". N can be viewed as a function from 2^{PL} to 2^{PL} such that for a set A of formulas, $N(A) = \{x \in PL : (a, x) \in N \text{ for some } a \in A\}$. Intuitively, N can be interpreted as a *normative code* composed of conditional norms and the set A serves as explicit input representing factual statements.

Makison and van der Torre [6] define the semantics of input/output logics from O_1 to O_4 as follows:

- $O_1(N, A) = Cn(N(Cn(A)))$.
- $O_2(N, A) = \bigcap\{Cn(N(V)) : A \subseteq V, V \text{ is complete}\}$.
- $O_3(N, A) = \bigcap\{Cn(N(B)) : A \subseteq B = Cn(B) \supseteq N(B)\}$.
- $O_4(N, A) = \bigcap\{Cn(N(V) : A \subseteq V \supseteq N(V)), V \text{ is complete}\}$.

Here Cn is the classical consequence operator of propositional logic, and a set of formulas is *complete* if it is either *maximal consistent* or equal to PL. These four operators are called *simple-minded output*, *basic output*, *simple-minded reusable output* and *basic reusable output* respectively. For each of these four operators, a *throughput* version that allows inputs to reappear as outputs, defined as $O_i^+(N, A) = O_i(N_{id}, A)$, where $N_{id} = N \cup \{(a, a) \mid a \in PL\}$. When A is a singleton, we write $O_i(N, a)$ for $O_i(N, \{a\})$.

Input/output logics are given a proof theoretic characterization. We say that an ordered pair of formulas is derivable from a set N iff (a, x) is in the least set that extends $N \cup \{(\top, \top)\}$ and is closed under a number of derivation rules. The following are the rules we need to define O_1 to O_4^+:

- SI (strengthening the input): from (a, x) to (b, x) whenever $b \vdash a$. Here $\vdash$ is the classical entailment relation of propositional logic.
- OR (disjunction of input): from (a, x) and (b, x) to $(a \vee b, x)$.
- WO (weakening the output): from (a, x) to (a, y) whenever $x \vdash y$.
- AND (conjunction of output): from (a, x) and (a, y) to $(a, x \wedge y)$.
- CT (cumulative transitivity): from (a, x) and $(a \wedge x, y)$ to (a, y).
- ID (identity): from nothing to (a, a).

The derivation system based on the rules SI, WO and AND is called D_1. Adding OR to D_1 gives D_2. Adding CT to D_1 gives D_3. The five rules together give D_4. Adding ID to D_i gives D_i^+ for $i \in \{1, 2, 3, 4\}$. $(a, x) \in D_i(N)$ is used to denote the norms (a, x) derivable from N using rules of derivation system D_i. In Makinson and van der Torre [6], the following soundness and completeness theorems are given:

Theorem 1 ([6]). *Given an arbitrary normative system N and formula a,*

- $x \in O_i(N, a)$ *iff* $(a, x) \in D_i(N)$*, for* $i \in \{1, 2, 3, 4\}$.
- $x \in O_i^+(N, a)$ *iff* $(a, x) \in D_i^+(N)$*, for* $i \in \{1, 2, 3, 4\}$.

2.2 Complexity theory

Complexity theory is the theory to investigate the time, memory, or other resources required for solving computational problems. In this subsection we briefly review those concepts and results from complexity theory which will be used in this paper. More comprehensive introduction of complexity theory can be found in [11, 1]

We assume the readers are familiar with notions like Turing machine and the complexity class P, NP and coNP. Oracle Turing machine and one complexity class related to oracle Turing machine will be used in this paper.

Definition 1 (oracle Turing machine). *An* oracle *for a language L is device that is capable of reporting whether any string w is a member of L. An (resp. non-deterministic) oracle Truing machine M^L is a modified (resp. non-deterministic) Turing machine that has the additional capability of querying an oracle. Whenever M^L writes a string on a special oracle tape it is informed whether that string is a member of L, in a single computation step.*

Definition 2 ($\mathbf{P}^{NP}$). *P^{NP} is the class of languages decidable with a polynomial time oracle Truing machine that uses oracle $L \in NP$.*

3 Complexity of input/output logic

The complexity of input/output logic has been sparsely studied in the past. Although the reversibility of derivations rules as a proof re-writing mechanism has been studied for input/output logic framework [6], the length or complexity of such proofs have not been developed. We approach the complexity of input/output logic from a semantic point of view.

We now start to study the complexity of the following input/output logics: O_1, O_1^+, O_3, and O_3^+. We focus on three different problems:
Given a finite set of norms N, a finite set of formulas A and a formula x:

(1) *Fulfillment problem:* is $x \in O(N, A)$?
(2) *Violation problem:* is $\neg x \in O(N, A)$?
(3) *Compatibility problem:* is $\neg x \notin O(N, A)$?

The aim of the fulfillment problem is to check whether the formula x appears among the obligations detached from the normative system N and facts A. The intuitive reading of the violation problem is: if the obligation to fulfill $\neg x$ exists, then x is a violation. Finally, the compatibility problem says if $\neg x$ is not obligatory, then x is compatible with the normative system N, given facts A. The compatibility problem is often referred as a *negative permission* [8, 3], and corresponds to what is called weak permission.[3] It can be proven that the other two problems can be reduced to the comliance problem. Therefore we focus on the compliance problem.

[3] "An act will be said to be permitted in the weak sense if it is not forbidden ..." [13].

3.1 Simple-minded O_1

Theorem 2. *The fulfillment problem of simple-minded input/output logic is coNP-complete.*

Corollary 1. *The violation problem of simple-minded input/output logic is coNP-complete. The compatibility problem of simple-minded input/output logic is NP-complete.*

3.2 Simple-minded throughput O_1^+

Theorem 3. *The fulfillment problem of simple-minded throughput input/output logic is coNP-complete.*

Corollary 2. *The violation problem of simple-minded throughput input/output logic is coNP-complete. The compatibility problem of simple-minded throughput input/output logic is NP-complete.*

3.3 Simple-minded reusable O_3

Theorem 4. *The fulfillment problem of simple-minded reusable input/output logic is between coNP and P^{NP}.*

Corollary 3. *The violation problem of simple-minded reusable input/output logic is between coNP and P^{NP}. The compatibility problem of simple-minded reusable input/output logic is between NP and P^{NP}.*

3.4 Simple-minded reusable throughput O_3^+

Theorem 5. *The fulfillment problem of simple-minded reusable throughput input/output logic is between coNP and P^{NP}.*

Corollary 4. *The violation problem of simple-minded reusable throughput input/output logic is between coNP and P^{NP}. The compatibility problem of simple-minded reusable throughput input/output logic is between NP and P^{NP}.*

4 Conclusion and future work

In this paper we develop complexity results of input/output logic. We show that four input/output logics (O_1, O_1^+, O_3, O_3^+) have lower bound coNP and upper bound either coNP or P^{NP}. There are several natural directions for future work:

1. What is the complexity of other input/output logic?
2. What is the complexity of constraint input/output logic? Constraint input/output logic [7] is developed to deal with the inconsistency of output. The semantics of constraint input/output logic is more complex than those input/output loic discussed in this paper. This might increase the complexity of the compliance problem. Constraint input/output logic based on O_3^+ has close relation with Reiter's default logic [10]. Gottlob [5] presents some complexity results of Reiter's default logic, which will give us insights on the complexity of constraint input/output logic.

3. What is the complexity of different types of permission? three different of permissions are introduced in Makinson and van der Torre [8]. In this paper we study the complexity of only one of them (namely, negative permissions) as the compatibility problem. The semantics of these three logics are different, which suggests different complexity for the new problems related to permissions.

References

1. Arora, S., Barak, B.: Computational Complexity: A Modern Approach. Cambridge University Press, New York, USA (2009)
2. Bochman, A.: A causal approach to nonmonotonic reasoning. Artificial intelligence 160(1-2), 105–143 (2004)
3. Boella, G., van der Torre, L.W.N.: Permissions and obligations in hierarchical normative systems. In: ICAIL. pp. 109–118 (2003), http://dl.acm.org/authorize?859450
4. Gabbay, D., Horty, J., Parent, X., van der Meyden, R., van der Torre, L. (eds.): Handbook of Deontic Logic and Normative Systems. College Publications, London (2013)
5. Gottlob, G.: Complexity results for nonmonotonic logics. J. Log. Comput. 2(3), 397–425 (1992), http://dx.doi.org/10.1093/logcom/2.3.397
6. Makinson, D., van der Torre, L.: Input-output logics. Journal of Philosophical Logic 29, 383–408 (2000)
7. Makinson, D., van der Torre, L.: Constraints for input/output logics. Journal of Philosophical Logic 30(2), 155–185 (2001)
8. Makinson, D., van der Torre, L.: Permission from an input/output perspective. Journal of Philosophical Logic 32, 391–416 (2003)
9. Parent, X., van der Torre, L.: I/O logic. In: Horty, J., Gabbay, D., Parent, X., van der Meyden, R., van der Torre, L. (eds.) Handbook of Deontic Logic and Normative Systems. College Publications (2013)
10. Reiter, R.: A logic for default reasoning. Artif. Intell. 13(1-2), 81–132 (1980), http://dx.doi.org/10.1016/0004-3702(80)90014-4
11. Sipser, M.: Introduction to the theory of computation. Cengage Learning, Boston, USA, 3 edn. (2012)
12. Sun, X.: How to build input/output logic. In: Bulling, N., van der Torre, L.W.N., Villata, S., Jamroga, W., Vasconcelos, W. (eds.) Computational Logic in Multi-Agent Systems - 15th International Workshop, CLIMA XV, Prague, Czech Republic, August 18-19, 2014. Proceedings. Lecture Notes in Computer Science, vol. 8624, pp. 123–137. Springer (2014), http://dx.doi.org/10.1007/978-3-319-09764-0_8
13. von Wright, G.: Norm and Action. Routledge and Kegan (1963)