UNIVERSITÄT ZU LÜBECK
INSTITUTE FOR SOFTWARE ENGINEERING
AND PROGRAMMING LANGUAGES

isp

# Seminar on Computational Contracting Approaches: Jørgensen's Dilemma

Karam Kharraz

November 27, 2020

# Outline

- Introduction
- Problem Definition: Jørgensen's Dilemma
- Proposed solution: I/O Logic
- Evaluation
- Conclusion

# Classification of Problems: Paper to be studied

| | | |
|---|---|---|
| 1. | Contrary-to-duty reasoning, preference and violation | preference |
| 2. | Non-deterministic actions: ought-to-do vs ought-to-be | agency |
| 3. | Moral luck and the driving example | agency |
| 4. | Procrastination: actualism vs possibilism | agency |
| 5. | Jørgensen's dilemma and the problem of detachment | norms |
| 6. | Multiagent detachment | norms |
| 7. | Coherence of a normative system | norms |
| 8. | Normative conflicts and dilemmas | preference & norms |
| 9. | Descriptive dyadic obligations and norms | preference & norms |
| 10. | Permissive norms | preference & norms |
| 11. | Meaning postulates and intermediate concepts | norms |
| 12. | Constitutive norms | norms |
| 13. | Revision of a normative system | norms |
| 14. | Merging normative systems | norms |
| 15. | Games, norms and obligations | norms & agency |

## Imperative and logic

Imperatives and Logic
Author(s): Jörgen Jörgensen
Source: *Erkenntnis (1930-1938)*, Vol. 7 (1937/1938), pp. 288-296
Published by: Springer

I. Imperative sentences are not capable of being either true or false. According to the logical positivist testability-criterion of meaning they therefore must be considered meaningless. However, they are nevertheless capable of being understood or misunderstood and seem also to be able to function as premisses as well as conclusions in logical inferences.

## Standard deontic Logic:1951

## Deontic logic: Axioms for Standard Deontic Logic [?] (System KD)

Set of axioms and inference rules to reason about a system of norms:

1. Taut: All tautologies From Propositional logic are wffs[1] of the language.
2. Axiom K: $O(p \rightarrow q) \rightarrow (O(p) \rightarrow O(q))$
3. Axiom D: $O(p) \rightarrow \neg O(\neg p)$
4. Modus Ponens:

$$\text{MP}\ \frac{\vdash p \rightarrow q \qquad \vdash p}{\vdash q}$$

5. Necessity:

$$\textsc{Nec}\ \frac{\vdash p}{\vdash O(p)}$$

[1] Well formed formulas

## Tautologies in Propositional logic

| p | q | (p → (p ∨ q)) |
|---|---|---|
| F | F | **T** |
| F | T | **T** |
| T | F | **T** |
| T | T | **T** |

- $p \to (p \vee q)$ is a tautology.
- In SDL using Necessity + K we get:

  $O(p) \to O(p \vee q)$

## Contrary to duty paradox caused by Jørgensen's Dilemma?

1. It is forbidden kill. $O(\neg kill)$
2. If one ever kills he must kill gently. $kill \rightarrow O(killgently)$.
3. Someone just killed. $kill$
4. $killgenly \rightarrow kill$ (Common sense)

by assuming the set

- from (2) and (3) By MP : $O(killgently)$
- we know that $killgently \rightarrow kill$.
- by NEC we get $O(killgently \rightarrow kill)$
- by Axiom K: $O(killgently \rightarrow kill) \rightarrow (O(killgently) \rightarrow O(kill))$
- by MP and Axiom D: we conclude: $\neg(O(\neg kill))$ (4)
- by (1) and (4) we have $O(\neg kill) \wedge \neg O(\neg kill)$ **Paradox (Gentle Murderer)**

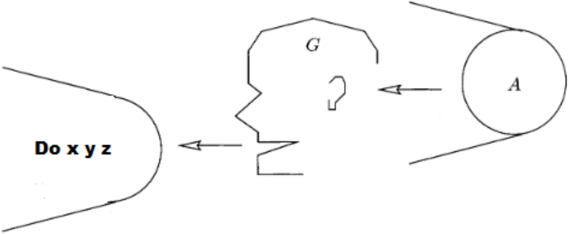
## Wrap up

- The Jørgensen's Trilemma could be understood as follows:
  1. Logical inference requires that the elements have truth-values.
  2. Normative statements do not have truth-values.
  3. There are logical inferences between normative statements.
- Proposed solution: *Input/Output logic*

## What is I/O Logic

- I/O logic is a non classical logic.
- Reason about a set $G$ of norms.
- Norms are pairs (a,x), where a and x are boolean formulae.
- $a$ is called an input, while $x$ is the corresponding output.
- The question is: What is the right output for a situation $A$ a set of boolean formulae.

# What should I do ? $Out_x(G, A)$



## Basic operations: Consequence

**Definition (Consequence Cn(A))**

$Cn(A)$ denotes the set of logical consequences of $A$ in classical propositional logic.It returns the set of all provable propositional formulae provable assuming the fact in $A$

$Cn(a) = \{x \mid A \vdash x\}$

**Example**

- $A_1 = \{x, y\}$ then $Cn(A) = \{x, y, x \vee y, x \wedge y \vee \ldots\}$

## Basic operations: Image of a $A$

### Definition (G(A))

$G(A)$ is the set of answers of a set of inputs.

$G(a) = \{x \mid (x, a).x \in G \wedge a \in A\}$

### Example

- $G_1 = \{(a_1, x_1), (a_2, x_2)\}$ and $A_1\{a_1, z\}$ and $A_2 = \{a_1, a_2, x_2\}$
- $G_1(A_1) = \{x_1\}$
- $G_1(A_2) = \{x_1, x_2\}$

## Derivable Properties

$$\top \; \frac{--}{\top, \top} \qquad \text{SI} \; \frac{(a,x) \quad b \vdash a}{(b,x)}$$

$$\text{WO} \; \frac{(a,x) \quad x \vdash y}{(a,y)} \qquad \text{AND} \; \frac{(a,x) \quad (a,y)}{(a, x \wedge y)}$$

$$\text{OR} \; \frac{(a,x) \quad (b,x)}{(a \vee b, x)} \qquad \text{CT} \; \frac{(a,x) \quad (a \wedge x, y)}{(a,y)}$$

| Output operation | Rules |
|---|---|
| Simple-minded ($out_1$) | {⊤, SI, WO, AND} |
| Basic ($out_2$) | {⊤, SI, WO, AND}+{OR} |
| Reusable ($out_3$) | {⊤, SI, WO, AND}+{CT} |
| Basic reusable ($out_4$) | {⊤, SI, WO, AND}+{OR,CT} |

## Derivable Properties

$$\top \ \frac{--}{\top, \top} \qquad \text{SI} \ \frac{(a, x) \quad b \vdash a}{(b, x)}$$

$$\text{WO} \ \frac{(a, x) \quad x \vdash y}{(a, y)} \qquad \text{AND} \ \frac{(a, x) \quad (a, y)}{(a, x \wedge y)}$$

$$\text{OR} \ \frac{(a, x) \quad (b, x)}{(a \vee b, x)} \qquad \text{CT} \ \frac{(a, x) \quad (a \wedge x, y)}{(a, y)}$$

| Output operation | Rules |
|---|---|
| Simple-minded ($out_1$) | {⊤, SI, WO, AND} |
| Basic ($out_2$) | {⊤, SI, WO, AND}+{OR} |
| Reusable ($out_3$) | {⊤, SI, WO, AND}+{CT} |
| Basic reusable ($out_4$) | {⊤, SI, WO, AND}+{OR,CT} |

## Contrary to duty reasoning in I/O Logic

### Definition (Maximal non excessive subsets)

$Maxfam(G, A, C) = H_{max} \subseteq G$ . $out_x(H, A)$ is consistent.

### Definition (Consistent out)

$out_x^{\cap}(G, A) = \bigcap \{out_x(H, A) | H \in Maxfam(G, A, A)\}$

### Example (Gentle murderer)

- $G = \{(\top, \neg kill), (kill, kill_gently)\}$and $A = \{kill\}$
- We have $H \in out_1(G, A) = \{kill\_gently, \neg kill\}$ inconsistent. because $\top$
- $H = \{kill\} \in out_1(G, A)$ and $Out(H, A) = kill\_gently$ is consistent
- $out_x^{\cap}(G, A) = kill\_gently$

# Conclusion

**Abstract.** We explain the *raison d'être* and basic ideas of input/output logic, sketching the central elements with pointers to other publications for detailed developments. The motivation comes from the logic of norms. Unconstrained input/output operations are straightforward to define, with relatively simple behaviour, but ignore the subtleties of contrary-to-duty norms. To deal with these more sensitively, we constrain input/output operations by means of consistency conditions, expressed via the concept of an outfamily. However, this is a more complex affair, with difficult choices between alternative options.