

Report from Dagstuhl Seminar 18171

Normative Multi-Agent Systems

Edited by

Mehdi Dastani¹, Jürgen Dix², Harko Verhagen³, and
Serena Villata⁴

1 Utrecht University, NL, m.m.dastani@uu.nl

2 TU Clausthal, DE, dix@tu-clausthal.de

3 Stockholm University, SE, verhagen@dsv.su.se

4 Laboratoire I3S – Sophia Antipolis, FR, villata@i3s.unice.fr

Abstract

This report documents the program and the outcomes of Dagstuhl Seminar 18171 “Normative Multi-Agent Systems”. Normative multi-agent systems combine models for multi-agent systems with normative concepts, like obligations, permissions, and prohibitions. As such, they promise to be a suitable model, for example for (regulated) multiagent societies, organizations, electronic institutions, autonomous agent cooperation (with humans-in-the-loop) and much more. The aim of this seminar was to bring together researchers from various scientific disciplines, such as computer science, artificial intelligence, philosophy, law, cognitive science and social sciences to discuss the emerging topic concerning the *responsibility* of autonomous systems. Autonomous software systems and multi-agent systems in open environments require methodologies, models and tools to analyse and develop flexible control and coordination mechanisms. Without them, it is not possible to steer the behaviour and interaction of such systems and to ensure important overall properties. *Normative multi-agent systems* is an established area focussing on how norms can be used to control and coordinate autonomous systems and multi-agents systems without restricting the autonomy of the involved systems. Such control and coordination systems allow autonomous systems to violate norms, but respond to norm violations by means of various sanctioning mechanisms. Therefore it is crucial to determine which agents or agent groups are accountable for norm violations. The focus of this seminar laid on how the responsibility of autonomous systems can be defined, modelled, analysed and computed.

Seminar April 22–27, 2018 – <http://www.dagstuhl.de/18171>

2012 ACM Subject Classification Computing methodologies → Multi-agent systems

Keywords and phrases autonomous systems, control and coordination, norm-based systems, responsibility

Digital Object Identifier 10.4230/DagRep.8.4.72

Edited in cooperation with Tobias Ahlbrecht

1 Executive Summary

Mehdi Dastani (Utrecht University, NL)

Jürgen Dix (TU Clausthal, DE)

Harko Verhagen (Stockholm University, SE)

License © Creative Commons BY 3.0 Unported license
© Mehdi Dastani, Jürgen Dix, and Harko Verhagen

The multi-disciplinary workshop on Normative Multi-Agent Systems attracted leading international scholars from different research fields (e.g. theoretical computer science, programming languages, cognitive sciences, law, and social sciences).



Except where otherwise noted, content of this report is licensed under a Creative Commons BY 3.0 Unported license

Normative Multi-Agent Systems, *Dagstuhl Reports*, Vol. 8, Issue 04, pp. 72–103

Editors: Mehdi Dastani, Jürgen Dix, Harko Verhagen, and Serena Villata



Dagstuhl Reports

Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

The seminar was a blend of talks, discussions and group work. It began on the first day with short “teaser talks” (10 + 5 minutes) related to the main topic of *norms and responsibility*, one given by almost each participant. The talks were meant to be inspiring and thought-provoking, channeling ideas for the following days. While some missed the established procedure with longer talks, the new format was overall very well received and allowed for many different thoughts and concepts to be presented and discussed in relatively short time.

Four working groups formed at the end of the first day for the norm-related topics responsibility, new logics, ethics/values and (machine) learning.

The aim of the group sessions, on the second and fourth day, was to get a shared understanding of the specific topics and to identify future research possibilities. Each group reported back in a plenary session at the end of each group work day, where the groups also tried to establish interconnections between them.

Responsibility. This group discussed how to grasp the very abstract concept of responsibility. A big chunk was dedicated to the formalization of responsibility. Many (vastly different) assumptions were laid out. The problem of “delegating responsibility” was discussed with special intensity. The group (being by far the largest one) split later to discuss different notions of responsibility on the basis of selected examples. A working paper was produced, included in this report under Section 4.1.

New logics. The aim of this group was to find out how to tackle norms and responsibility in terms of logics, especially how new logics for this task could be devised.

Ethics/values. This group discussed the more ethics-oriented aspects of normative systems. Values provide an additional layer for normative reasoning: e.g. “how acceptable is it to violate a given norm?” The group produced a draft of a paper on “The Value(s) of Water” connecting NorMAS to the AI for Good initiative. Work is planned to continue during 2018 resulting in a paper for publication, e.g. in ACM communications or a similar outlet.

(Machine) Learning. The learning group discussed the opportunity of integrating norms and responsibility into machine learning procedures. As those are usually opaque, this presents as a notable challenge. For example, the learning’s input data has to be pre-processed to get a normatively acting system. Also, the learned sub-symbolic system should be enhanced with “regular” symbolic reasoning, which can be better regulated by norms and analysed for responsibility.

The fourth day was further enriched by a brainstorming session to identify possible applications. The subsequent clustering revealed the topics

- **transport**, e.g. smart grid/home, intelligent cars,
- **tools**, e.g. for autonomous service composition, legal reasoning, or supporting software/requirements engineering,
- **climate & agriculture**, e.g. agents negotiating fertilizer and water use, or an app that helps monitoring personal climate-affecting activities,
- **societies**, e.g. norms improving sustainability, monitoring of online forums for bad behavior or hate speech detection,
- **security**, e.g. protecting personal freedom by dynamically analysing normative consequences of law proposals, monitoring a company’s compliance with EU regulations, improving access to restricted access datasets, or making societies resilient for data surveillance by means of contract negotiations,

- **health**, e.g. ethical decision-making, norms for improving personal health and fitness, defining wellbeing by norms, handling of patient/health data, and a big interest in healthcare robots,
- **energy**, e.g. modelling energy security with norms, managing air quality, observing long-term consequences, agents monitoring (personal) energy use to identify bad behavior, or regulating industrial relations or the energy and material footprint.

The application areas were discussed in a plenary session and formed the input to the discussion on future plans for the NorMAS community. Several conferences were identified to target proposals for a NorMAS-related workshop as part of the event. The community sees many relevant application areas not in the least in autonomous internet services and physical agents such as robots, vehicles and drones, where social reasoning will be of the utmost importance. Bringing the work from NorMAS to these areas will be highly beneficial to the involved communities.

2 Contents

Executive Summary

Mehdi Dastani, Jürgen Dix, and Harko Verhagen 72

Overview of Talks

Norms in the Multi-Agent Programming Contest
Tobias Ahlbrecht 76

Causality, Responsibility and Blame in Team Plans
Natasha Alechina, Joseph Halpern, and Brian Logan 76

Overview of Legal Liability of Autonomous Systems and Implications for Norms-based Systems
Kevin D. Ashley 76

On the role of accountability in programming MAS
Matteo Baldoni, Cristina Baroglio, and Roberto Micalizio 77

A Formalisation of Moral Responsibility and the Problem of Many Hands
Tiago de Lima 79

Supervising Autonomous Systems
Davide Dell’Anna 80

Isabelle/HOL: a Computational Framework for Normative Reasoning
Ali Farjami, Christoph Benzmüller, and Xavier Parent 80

Natural Strategic Ability
Wojtek Jamroga 81

Programming Responsibility in Norm-Aware Agents
Brian Logan 81

Simulating the hermeneutics of irresponsibility
Martin Neumann 81

Anchoring Electronic Institutions
Pablo Noriega, Julian Padget, and Harko Verhagen 82

Rule Based SLAs for Water (RBSLA4Water)
Adrian Paschke 83

Goal-based Argumentation for Intelligent Deliberation
Douglas Walton 84

Trust, Responsibility, and Explanation
Michael Winikoff 84

Group Responsibility Under Imperfect Information
Vahid Yazdanpanah 85

Working groups

Formal definitions of responsibility
Natasha Alechina, Tiago de Lima, Brian Logan, Ken Satoh, and Douglas Walton . 85

Participants 103

3 Overview of Talks

3.1 Norms in the Multi-Agent Programming Contest

Tobias Ahlbrecht (TU Clausthal, DE)

License © Creative Commons BY 3.0 Unported license
© Tobias Ahlbrecht

I briefly present the Multi-Agent Programming Contest, a competition attempting to stimulate research in the area of multi-agent system development and programming. I will touch on its potential for norm usage and evaluation and vice versa, with regard to the opportunity of incorporating norms in the next scenario.

3.2 Causality, Responsibility and Blame in Team Plans

Natasha Alechina (University of Nottingham, GB), Joseph Halpern, and Brian Logan (University of Nottingham, GB)

License © Creative Commons BY 3.0 Unported license
© Natasha Alechina, Joseph Halpern, and Brian Logan

Joint work of Natasha Alechina, Joseph Halpern, Brian Logan

Main reference Natasha Alechina, Joseph Halpern, Brian Logan: “Causality, Responsibility and Blame in Team Plans”, in Proc. of the 16th Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2017, São Paulo, Brazil, May 8-12, 2017, pp. 1091–1099, ACM, 2017.

URL <http://dl.acm.org/citation.cfm?id=3091279>

Many objectives can be achieved (or may be achieved more effectively) only by a group of agents executing a team plan. If a team plan fails, it is often of interest to determine what caused the failure, the degree of responsibility of each agent for the failure, and the degree of blame attached to each agent. In the talk, I will show how team plans can be represented in terms of structural equations, and how the definitions of causality introduced by Halpern (2015) and degree of responsibility and blame introduced by Chockler and Halpern (2004) can be applied to determine the agent(s) who caused the failure and what their degree of responsibility/blame is. I will present results on the complexity of computing causality and degree of responsibility and blame, which show that they can be determined in polynomial time for many team plans of interest. The talk is based on joint work with Joseph Halpern and Brian Logan.

3.3 Overview of Legal Liability of Autonomous Systems and Implications for Norms-based Systems

Kevin D. Ashley (University of Pittsburgh, US)


License © Creative Commons BY 3.0 Unported license
© Kevin D. Ashley

Autonomous systems present novel circumstances for assessing legal liability. Autonomous vehicles, for instance, promise to increase traffic safety overall. Inevitably, however, such vehicles will also cause accidents injuring people and property, and the providers of such vehicles and their component software systems will be subject to law suits on behalf of victims. This talk briefly surveys how the American law of product liability and negligence

would address such scenarios and highlights some potentially interesting practical and legal differences between a machine learning versus a norms-based architecture when autonomous vehicles cause accidents. The legal framework could lead to a discussion to elicit more details about the norms-based and machine learning architectures in order to explore in greater depth these potential practical and legal differences where the ML-based perceptual system and the norms-based reasoner meet.

3.4 On the role of accountability in programming MAS

Matteo Baldoni (University of Turin, IT), Cristina Baroglio, and Roberto Micalizio

License  Creative Commons BY 3.0 Unported license
© Matteo Baldoni, Cristina Baroglio, and Roberto Micalizio

Multiagent Systems (MAS) represents a viable programming paradigm for the development of complex systems characterized by multiple threads of execution that run in parallel. Most of the design methodologies and programming platforms that have been proposed in the literature (e.g., OperA [8], OMNI [9], OCEAN [12], 2OPL [7], JaCaMo [3], and [11]) are grounded on the metaphor of the *organization*: The system under development is seen as a human-like organization where organizational goals, possibly decomposed into subgoals, are distributed to agents playing organizational roles. A set of norms rule the admissible interactions among agents within the organization. Such a normative system issues obligations, permissions, and prohibitions as a consequence of what agents do within the organization. Notably, obligations and the like do not require any acceptance by the agents. Indeed, obligations are the means through which an organization stimulates the agents to perform some tasks. Of course, agents, inasmuch autonomous entities, can decide whether to satisfy an obligation or violate a prohibition. Thus, in order to enforce the norm-specified, desired behavior some sanctioning mechanism is often introduced. The idea is that a rational agent will satisfy obligations to avoid sanction.

We deem that the organizational metaphor is a very effective way to approach the design and development of complex systems, but the current formalizations are still incomplete in properly capturing the notion of organization from a software engineering point of view. Current approaches, in fact, strongly depend on *obligations* for getting tasks done, but this imposes some, often unspoken, assumptions. First, since an obligation towards an agent is satisfied when the agent activates a proper behavior, it is assumed that the agent has necessarily a proper behavior for each obligation it will ever receive. This assumption is easily satisfied only when the set of goals that can be assigned to an agent are known in advance and do not change over time. But goals are dynamic by nature, and hence it may be possible that when the normative system issues an obligation towards an agent, that agent does not have a proper behavior for satisfying that obligation. We have demonstrated this in the context of JaCaMo platform (see [2]). Second, it is assumed that the sanctions associated with the violation of an obligations are a sufficient tool for conditioning agents' behaviors. Agents, however, can deliberately decide to violate an obligation despite the sanction, and will do so in those cases when the obligation does not match the agent's goals and the sanction is acceptable. Thus, obligations may fall short in stimulating agents doing tasks, either because they can be directed to agents that do not possess the proper capabilities, or because the sanction is not an absolute criteria for an agent to decide how to act.

It is interesting to note that such shortcomings of obligations are well-known and widely accepted in sociology (see, for instance, [10, 15, 13]). In social terms, an agent *voluntarily* triggers an act only if that act is *desirable* for the agent. Therefore, normative sanctions often have little consequence on the agent, and no consequence at the society level. It was also observed in the requirements engineering field [6] that agents' obedience to the system norms cannot be taken for granted. Agent autonomy demands a different way of conceptualizing software modularity: not in terms of subgoals that are assigned to the agents, but rather in terms of responsibilities that are explicitly taken on by the agents. This last observation concerns also approaches that, instead of relying on norms/obligations, rely on social commitments [5, 16]. On the one side, the creation of a commitment is a deliberate act of the agent that takes on a duty. On the other side, however, a detached commitment is a directed obligation from the debtor to the creditor of the commitment. As such, an agent can violate its commitments when it deems advantageous to do so. We deem that a commitment is still inadequate for modeling "responsibilities" in a way that can be exploited from a software engineering perspective. In fact, agents could create commitments to bring about conditions that are not completely under their control. In these cases, sanctioning an agent that has not satisfied a commitment is of little help.

In this paper we argue that the current models for supporting agent interaction and coordination – norm/obligation-oriented, as well as commitment-oriented – should be complemented in some way. We found support to our intuition in the literature from the areas of sociology (and in particular ethnomethodology) and from political sciences, identifying in *accountability* the key missing concept. Starting from sociology, citing [4]: "Garfinkel developed the idea of the accountable character of action to emphasize that social action is organized so that it can be reported and described. In other words, *people design social actions so that others can see and say what those actions are*. For ethnomethodologists, accountability is a pervasive feature of how people co-ordinate their actions." Instead, from political sciences [1]: if an individual is accountable, that accountability will act as a constraint on their decision process. Holding people accountable means asking them to explain their actions, especially when they fail to bring about expected goals. Accountability is therefore the underlying force that influence actions in human organizations, and more generally, in human relationships.

On the software side, accountability can be a powerful tool for motivating better practices, and consequently more reliable and trustworthy systems [14]. Our intuition is that accountability can be understood as a software engineering element that helps a designer devise a complex system and, at the same time, can be the base for handling exceptions at run time in a more effective way than of an obligation/sanction mechanism. In this ongoing work we explore the possibility to found the realization of distributed systems on the two basic notions of responsibility and accountability, tracing connecting points with more traditional approaches, and tracing also directions of research that we deem significant.

References

- 1 Paul A. Anderson. Justifications and precedents as constraints in foreign policy decision-making. *American Journal of Political Science*, 25(4), 1981.
- 2 Matteo Baldoni, Cristina Baroglio, Katherine M. May, Roberto Micalizio, and Stefano Tedeschi. ADOPT JaCaMo: Accountability-Driven Organization Programming Technique for JaCaMo. In A. Bo, A. Bazzan, J. Leite, L. van der Torre, and S. Villata, editors, *PRIMA 2017: Principles and Practice of Multi-Agent Systems, 20th International Conference*, number 10621 in Lecture Notes in Computer Science, pages 295–312. Springer, 2017.
- 3 Olivier Boissier, Rafael H. Bordini, Jomi F. Hübner, Alessandro Ricci, and Andrea Santi. Multi-agent oriented programming with JaCaMo. *Sci. Comput. Program.*, 78(6):747–761, 2013.

- 4 Graham Button and Wes Sharrock. The organizational accountability of technological work. *Social Studies of Science*, 28(1):73–102, 1998.
- 5 C. Castelfranchi. Principles of Individual Social Action. In *Contemporary action theory: Social action*, volume 2, pages 163–192, Dordrecht, 1997. Kluwer.
- 6 Amit K Chopra and Munindar P Singh. From social machines to social protocols: Software engineering foundations for sociotechnical systems. In *Proc. of the 25th Int. Conf. on WWW*, 2016.
- 7 Mehdi Dastani, Nick AM Tinnemeier, and John-Jules Ch Meyer. A programming language for normative multi-agent systems. In *Handbook of Research on Multi-Agent Systems: semantics and dynamics of organizational models*, pages 397–417. IGI Global, 2009.
- 8 Virginia Dignum. *A model for organizational interaction: based on agents, founded in logic*. PhD thesis, Utrecht University, 2004. Published by SIKS.
- 9 Virginia Dignum, Javier Vázquez-Salceda, and Frank Dignum. OMNI: introducing social structure, norms and ontologies into agent organizations. In *Programming Multi-Agent Systems, Second International Workshop ProMAS, Selected Revised and Invited Papers*, volume 3346 of *Lecture Notes in Computer Science*, pages 181–198. Springer, 2004.
- 10 Emile Durkheim. *De la division du travail social*. 1893.
- 11 Marc Esteva, Juan-Antonio Rodríguez-Aguilar, Carles Sierra, Pere Garcia, and Josep L. Arcos. On the formal specification of electronic institutions. In Frank Dignum and Carles Sierra, editors, *Agent Mediated Electronic Commerce: The European AgentLink Perspective*, pages 126–147. Springer Berlin Heidelberg, Berlin, Heidelberg, 2001.
- 12 Nicoletta Fornara, Francesco Viganò, Mario Verdicchio, and Marco Colombetti. Artificial institutions: a model of institutional reality for open multiagent systems. *Artificial Intelligence and Law*, 16(1):89–105, 2008.
- 13 Harold Garfinkel. *Studies in ethnomethodology*. Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1967.
- 14 Helen Nissenbaum. Accountability in a computerized society. *Science and engineering ethics*, 2(1):25–42, 1996.
- 15 Talcott Parsons. *The Structure of Social Action*. Collier-Macmillan, London, 1968.
- 16 Munindar P. Singh. An ontology for commitments in multiagent systems. *Artif. Intell. Law*, 7(1):97–113, 1999.

3.5 A Formalisation of Moral Responsibility and the Problem of Many Hands

Tiago de Lima (CNRS - Lens, FR)

License © Creative Commons BY 3.0 Unported license
© Tiago de Lima

In this talk, I present a formalization of the so called problem of many hands. Using the basic concepts upon which the meanings of responsibility are defined, we construct a logic which enables us to express sentences like ‘individual i is accountable for ϕ ’, ‘individual i is blameworthy for ϕ ’ and ‘individual i has the obligation to see to it that ϕ ’. Such effort contributes to the discussion about responsibility in at least two ways. First, it clarifies the definitions and also their differences and similarities. Second, it assesses the consistency of the formalization of responsibility, not only by showing that definitions are not inconsistent, but also by providing a formal demonstration of the relation between three different meanings of the word responsibility. Moreover, the formal account can be used to derive new properties of the concepts, thus, giving new insights that can be used to advance the discussion. And

finally, the formalism proposed here provides a framework wherein criteria for ascribing responsibilities can be stated and, if individuals are to be held responsible for outcomes, then, at least, justifications can be made clear.

3.6 Supervising Autonomous Systems

Davide Dell’Anna (Utrecht University, NL)

License © Creative Commons BY 3.0 Unported license
© Davide Dell’Anna

Joint work of Davide Dell’Anna, Mehdi Dastani, Fabiano Dalpiaz

Norms with sanctions have been widely employed as a mechanism for controlling and coordinating the behavior of agents without limiting their autonomy. The norms enforced in a multi-agent system (MAS) can be revised in order to increase the likelihood that desirable system properties (such as company’s core values or ethical principles) are fulfilled or that system performance is sufficiently high. We provide a description of a supervision system that monitors the execution of a MAS, identifies deviations from the overall system objectives, and with the help of a probabilistic model (Bayesian Network) automatically proposes norm revisions that are expected to increase system objectives achievement. A preliminary experimental evaluation of the effectiveness of the framework on an urban smart transportation simulator is proposed. The experimental results are promising: data retrieved from system execution can be successfully employed to suggest and apply appropriate revisions of norms at runtime, allowing the MAS to reach an adequate satisfaction of the desired overall system objectives.

3.7 Isabelle/HOL: a Computational Framework for Normative Reasoning

Ali Farjami (University of Luxembourg, LU), Christoph Benzmüller (FU Berlin, DE), and Xavier Parent

License © Creative Commons BY 3.0 Unported license
© Ali Farjami, Christoph Benzmüller, and Xavier Parent

Joint work of Christoph Benzmüller, Ali Farjami, Xavier Parent

Main reference Christoph Benzmüller, Ali Farjami, Xavier Parent: “Faithful Semantical Embedding of a Dyadic Deontic Logic in HOL”, CoRR, Vol. abs/1802.08454, 2018.

URL <http://arxiv.org/abs/1802.08454>

We have provided the theoretical foundation for the implementation and automation of dyadic deontic logic within off-the-shelf higher-order theorem provers and proof assistants. We have devised (shallow) semantical embedding of some dyadic deontic logics in classical higher-order logic. The embedding has been encoded in Isabelle/HOL, which turns this system into a proof assistant for deontic logic reasoning. The experiments with this environment provide evidence that these logic implementations fruitfully enables interactive and automated reasoning at the meta-level and the object-level. We built a computational framework, based Isabelle/HOL, for normative reasoning.

3.8 Natural Strategic Ability

Wojtek Jamroga (Polish Academy of Sciences - Warsaw, PL)

License © Creative Commons BY 3.0 Unported license
© Wojtek Jamroga

Joint work of Wojtek Jamroga, Vadim Malvone, Aniello Murano
Main reference Wojtek Jamroga, Vadim Malvone, Aniello Murano: “Reasoning about Natural Strategic Ability”, in Proc. of the 16th Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2017, São Paulo, Brazil, May 8-12, 2017, pp. 714–722, ACM, 2017.

URL <http://dl.acm.org/citation.cfm?id=3091227>

In game theory, as well as in the semantics of game logics, a strategy can be represented by any function from states of the game to the agent’s actions. That makes sense from the mathematical point of view, but not necessarily in the context of human behavior. This is because humans are quite bad at executing complex plans, and also rather unlikely to come up with such plans in the first place. In this work, we adopt the view of bounded rationality, and look only at "simple" strategies in specifications of agents’ abilities. I will formally define what "simple" means, and present a variant of alternating-time temporal logic that takes only such strategies into account. I will also briefly point out where it possibly connects with the notion of responsibility.

3.9 Programming Responsibility in Norm-Aware Agents

Brian Logan (University of Nottingham, GB)

License © Creative Commons BY 3.0 Unported license
© Brian Logan

In this talk I will consider the problem of programming group norms specifying that a group of agents are responsible for bringing about some state, i.e., a group obligation. The group norm specifies what should be achieved, by when, and the sanction for the group if the norm is violated, but not the responsibilities of each agent to bringing about the desired state or individual sanctions in the event of a violation. As such they provide a degree of abstraction that is critical for the implementation of many large normative MAS. However group norms introduce several new programming challenges, in particular the delegation of responsibility for norm enforcement from the MAS to an agent or agents within the group. I will present an approach to implementing group-norm-aware agents that are able to deliberate on their individual goals, group norms and sanctions when deciding whether to participate in a team plan.

3.10 Simulating the hermeneutics of irresponsibility

Martin Neumann (Jacobs University Bremen, DE)

License © Creative Commons BY 3.0 Unported license
© Martin Neumann

In the talk I will approach the issue of responsibility from the reverse angle by investigating corruption as a manifestation of irresponsibility. Corruption is a phenomenon of misuse of a position of trust. As case the Ukraine is selected, which is characterized by a current high level of corruption. The project addresses the question of how civil society can be organized in the

interplay of political and legislative institutions and cultural dimensions of civil engagement. Addressing the perception of (ir)responsible fulfillment of social roles during interactions need to take a cultural dimension into account. This requires socio-cognitive coupling of how participants make sense of the phenomenology of a situation from the perspective of their worldview. For this purpose a methodology will be applied that has been developed in the previous project GLODERS integrating qualitative content analysis, agent-based simulation, and narrative analysis of simulation results. Central feature is preserving traceability to the empirical evidence throughout the research process. Traceability enables interpretation of simulations by generating a narrative storyline of the simulation. Thereby simulation enables a qualitative exploration of textual data. The whole process generates a thick description of the subject of study. Simulation results generate virtual narratives by decomposing and rearranging the empirical in-vivo codes. This can be described as an exploration of the horizon of the space of cultural possibilities. The talk will outline work in progress and I hope for stimulating feedback at an early stage of research.

3.11 Anchoring Electronic Institutions

Pablo Noriega (IIIA - CSIC - Barcelona, ES), Julian Padget (University of Bath, GB), and Harko Verhagen (Stockholm University, SE)

License © Creative Commons BY 3.0 Unported license

© Pablo Noriega, Julian Padget, and Harko Verhagen

Joint work of Pablo Noriega, Harko Verhagen, Mark d’Inverno, Julian Padget

Main reference Pablo Noriega, Harko Verhagen, Mark d’Inverno, Julian Padget: “A Manifesto for Conscientious Design of Hybrid Online Social Systems”, in Proc. of the Coordination, Organizations, Institutions, and Norms in Agent Systems XII - COIN 2016 International Workshops, COIN@AAMAS, Singapore, Singapore, May 9, 2016, COIN@ECAI, The Hague, The Netherlands, August 30, 2016, Revised Selected Papers, Lecture Notes in Computer Science, Vol. 10315, pp. 60–78, Springer, 2016.

URL http://dx.doi.org/10.1007/978-3-319-66595-5_4

Online Institutions capture the three main features that characterise classical institutions: (i) they are a set of artificial constraints that articulate human interactions (North); (ii) they are a regulated social space where institutional actions and facts take place (Searle); and (iii) they are coordination artefacts that constitute an interface between the individual decision-making models of agents and a collective activity they pursue (Simon). They can be understood as socio-cognitive technical systems in as much as all interactions happen online, and the agents that participate in them may be natural or artificial entities endowed with some form of social rationality. Moreover they are normative multiagent systems because, actually, only those interactions that comply with –enforced– institutional norms may have an institutional effect.

In this paper we are concerned with a very practical problem: what one has to take into account so that an online institution works effectively in the real world (in the sense that attempted actions, only when deemed institutionally admissible, produce the actual intended effects). We approach this question in two steps: first we discuss how an abstract isolated institution may be anchored and then we extend the discussion to institutions that are situated in a wider and changing socio-technical environment.

For our discussion we build on the “WIT framework” that represents an institution as three interconnected views (working, institutional and technological), and the requirements for “conscientious” design (thoroughness, mindfulness and responsibility) [1].

The use of the WIT framework allows for a separation of concerns implicit in the design and implementation of a given electronic institution. Thus we inspect the pragmatical

requirements of the three views and their pair-wise relationships, and elucidate what needs to be satisfied in order to guarantee that the online institution functions properly.

References

- 1 Pablo Noriega, Harko Verhagen, Mark d’Inverno, and Julian Padget. A manifesto for conscientious design of hybrid online social systems. In Stephen Craneffeld, Samhar Mahmoud, Julian Padget, and Ana Paula Rocha, editors, *Coordination, Organizations, Institutions, and Norms in Agent Systems XII - COIN 2016 International Workshops, COIN@AAMAS, Singapore, Singapore, May 9, 2016, COIN@ECAI, The Hague, The Netherlands, August 30, 2016, Revised Selected Papers*, volume 10315 of *Lecture Notes in Computer Science*, pages 60–78. Springer, 2016.

3.12 Rule Based SLAs for Water (RBSLA4Water)

Adrian Paschke (FU Berlin, DE)

License © Creative Commons BY 3.0 Unported license
© Adrian Paschke

Joint work of George Iordache, Adrian Paschke, Mariana Mocanu, Catalin Negru

Main reference George Iordache, Adrian Paschke, Mariana Mocanu, Catalin Negru: “Service Level Agreement Characteristics of Monitoring Wireless Sensor Networks for Water Resource Management (SLAs4Water)”, in *SIC Journal (Studies in Informatics and Control)*, Special Issue Advanced Services in Heterogeneous Distributed Systems, vol. 26(4), pp. 379–386, 2017

URL <https://doi.org/10.24846/v26i4y201701>

Water monitoring infrastructures use various components such as supervisory, control and data acquisition systems, wireless sensors or smart meters producing data in different formats and scales. [1] One of the most important characteristics of a Service Level Agreement (SLA) or executable Smart Contract when discussing about Monitoring Wireless Sensor Networks (MWSNs) is its effectiveness in assuring business success, a high provider profit, an increased level of client satisfaction and trust. In order to ensure that these goals will be achieved the provider of the MWSN must define several parameters [2] that characterize the Service Level Agreement between the MWSN provider and the MWSN customer. The characteristics of the SLA in place between the MWSN provider and the MWSN customer must be defined by taking into consideration various parameters that are particular to the MWSN such as routing algorithms, recovery from failure, monitoring and reporting aspects. This talk addresses a solution for an efficient and effective Service Level Agreement (SLA) design [3] and an implementation that applies a Rule-based SLA (RBSLA) solution [4], implemented by distributed Provalet agents [5], for the automated monitoring and enforcement of the service level objectives in the case of water resources management. The underlying logic applies the ContractLog knowledge representation [4, 6] and the Rule Based Service Level Agreement RuleML language [7].

References

- 1 EU H2020 Data4Water project - D1.1 Technology survey: Prospective and challenges. <http://data4water.pub.ro/mod/book/tool/print/index.php?id=104>, accessed Feb. 2018.
- 2 Paschke, A., Schnappinger-Gerull, E. A Categorization Scheme for SLA Metrics. Multi-Conference Information Systems (MKWI06), Passau, Germany, 2006.
- 3 George Iordache, Adrian Paschke, Mariana Mocanu and Catalin Negru. Service Level Agreement Characteristics of Monitoring Wireless Sensor Networks for Water Resource Management (SLAs4Water). In *SIC Journal (Studies in Informatics and Control)*, Special Issue Advanced Services in Heterogeneous Distributed Systems, 11/2017.

- 4 Adrian Paschke, Martin Bichler. Knowledge representation concepts for automated SLA management. *Decision Support Systems*, 46(1): 187-205 (2008)
- 5 Adrian Paschke. Provalets – Component-based Mobile Agents for Rule-based Data Access, Processing and Analytics. In Special Issue on Linked Data in Business in *Journal of Business & Information Systems Engineering (BISE)*, 5/2016.
- 6 Paschke, A., Bichler, M., Dietrich, J. ContractLog: An Approach to Rule Based Monitoring and Execution of Service Level Agreements. *International Conference on Rules and Rule Markup Languages for the Semantic Web (RuleML 2005)*, Galway, Ireland, 2005.
- 7 Adrian Paschke. RBSLA – Rule based Service Agreements. *Rule-based Contract Representation and Management for electronic Contracts, Policies and Service Level Agreements*, <http://rbsla.ruleml.org/>, accessed Feb. 2018.

3.13 Goal-based Argumentation for Intelligent Deliberation

Douglas Walton (University of Windsor, CA)

License © Creative Commons BY 3.0 Unported license
© Douglas Walton

Joint work of Douglas Walton, Alice Toniolo, Timothy J. Norman

Main reference Douglas Walton, Alice Toniolo, Timothy J. Norman: “Towards a richer model of deliberation dialogue: Closure problem and change of circumstances”, *Argument & Computation*, Vol. 7(2-3), pp. 155–173, 2016.

URL <http://dx.doi.org/10.3233/AAC-160009>

This paper surveys some recent work in argumentation theory on the problem how a group of autonomous intelligent agents can use goal-based defeasible reasoning in a normative dialogue setting to arrive rationally at a conclusion on what is the best thing to do to do in a changing set of circumstances requiring action. Resources from argumentation studies (an interdisciplinary field) are shown to be useful for current research of how to model arguments about responsibility in multiagent systems. Argumentation-based models of intelligent deliberation dialogue are shown to be useful for developing autonomous systems that support human practical (goal-based) reasoning. Having an open knowledge base enabling new evidence to be taken in as the deliberation proceeds is shown to be an important feature if the system is to model realistic deliberation.

3.14 Trust, Responsibility, and Explanation

Michael Winikoff (University of Otago, NZ)


License © Creative Commons BY 3.0 Unported license
© Michael Winikoff

Joint work of Michael Winikoff, Virginia Dignum, Frank Dignum

My talk considered the overarching issue of trusting autonomous systems, and the factors that lead to appropriate levels of trust in autonomous systems. I particularly focussed on the role of explanation, and described an explanation mechanism and its evaluation.

3.15 Group Responsibility Under Imperfect Information

Vahid Yazdanpanah (University of Twente, NL)

License  Creative Commons BY 3.0 Unported license
© Vahid Yazdanpanah

Joint work of Vahid Yazdanpanah, Mehdi Dastani, Wojciech Jamroga

A major issue in autonomous multi-agent systems is to determine who bears the responsibility for avoiding the occurrence of undesirable events. In this work, we take a forward-looking approach and model responsibility based on agents' preclusive power with respect to a given state of affairs. While some recent contributions tackled the issue under the perfect information assumption, we look at the broader picture, and provide operational semantics for reasoning about responsibility under imperfect information.

4 Working groups

4.1 Formal definitions of responsibility

Natasha Alechina (University of Nottingham, GB), Tiago de Lima (CNRS - Lens, FR), Brian Logan (University of Nottingham, GB), Ken Satoh (National Institute of Informatics - Tokyo, JP), and Douglas Walton (University of Windsor, CA)

License  Creative Commons BY 3.0 Unported license
© Natasha Alechina, Tiago de Lima, Brian Logan, Ken Satoh, and Douglas Walton

4.1.1 Introduction

The aim of this working paper is to investigate how responsibility may be formalised.¹ We consider four formalisms, modal logic, a logic of strategic ability, causal models and formal arguments, and for each formalism, we show how responsibility for an event or state of affairs can be formalised in two simple settings. We focus on responsibility for violations of a norm, specifically responsibility for failure to discharge an obligation.

4.1.2 The simplest case

We begin by considering the simplest case, where an agent is obliged to perform an action² and only that agent acts.

► **Example 1.** A plant must be watered in order to prevent it dying. Agent 1 has an obligation to water the plant. Agent 1 does not water the plant. The plant dies. Who is responsible for the death of the plant? Who is responsible for the violation of the obligation?

In this simple setting, responsibility for the state of affairs, and responsibility for violation of the norm coincide. In the remainder of this section, we show how responsibility can be modelled in each of the four formalisms we consider.

¹ This working paper can be seen as the report of a working group on formalising responsibility in normative multi-agent systems that formed part of Dagstuhl Seminar 18171 *Normative Multi-Agent systems* held at Schloss Dagstuhl in April 2018.

² Or bring about a state of affairs; in this simple example, where there is a single action that brings about a state of affairs, the two notions coincide.

4.1.2.1 Modal logic with three modalities [12]

In this section, we formalise responsibility in a modal logic with three modalities: knowledge K , obligation O and possibility (executability of an action) \diamond . Essentially we need to be able to say that the agent knows that it has an obligation to keep the plant alive, it knows causal dependency between watering and the plant being alive, and it knows that it is able to water the plant. Knowledge is veridical, so the statements the agent knows indeed hold. The following statements describe legal requirements for the intensional responsibility of agent 1 for the death of the plant, and the violation of the obligation (w is watering, d is plant is dead):

facts $w \rightarrow \neg d, \neg w, d$ (objective causality and objective facts)³

knowledge of obligation $K(O\neg d)$

knowledge of capability $K\diamond w$

understanding of what the agent is doing $K\neg w$

knowledge of causality $K(w \rightarrow \neg d)$

Unintentional violation (error/negligence): when instead of knowledge of causality $K(w \rightarrow \neg d)$ we have $O(K(w \rightarrow \neg d)) \wedge \neg K(w \rightarrow \neg d)$.

4.1.2.2 Logic of strategic ability [3]

In this section, we consider the formalism Coalition Epistemic Dynamic Logic (CEDL) as proposed in [4, 3]. It is a propositional multi-modal logic whose language is built using a countable set \mathbb{P} of propositional variables, a finite set \mathbb{N} of agent names and a finite set \mathbb{A} of action names. A joint action is defined as a total function $\delta : \mathbb{N} \rightarrow \mathbb{A}$. A partial joint action $\delta|_G$ is defined as the set $\{(i, a) \mid i \in G \text{ and } (i, a) \in \delta\}$. In addition to the usual connectives \neg and \wedge , the logic also has a modal operator for knowledge and another one for actions. A formula of the form $K_G\varphi$ means ‘the group of agents G knows that φ ’ (distributive knowledge). A formula of the form $[\delta]\varphi$ means ‘after all possible executions of δ , it is the case that φ ’. In this case, the idea is that each agent in \mathbb{N} execute its corresponding action in δ simultaneously. The language also permits the use of partial joint actions $a|_G$. Thus, a formula of the form $[a|_G]\varphi$ is also possible. In this case the idea is that each agent in G execute its corresponding action in δ simultaneously and we do not consider what the other agents in $\mathbb{N} \setminus G$ are doing. We have as its meaning ‘after all possible executions of $a|_G$ by the group of agents G and whatever the agents in $\mathbb{N} \setminus G$ do, it is the case that φ ’.

The models of this logic are structures of the form $M = \langle W, \{R_i \mid i \in \mathbb{N}\}, \{T_\delta \mid \delta : \mathbb{N} \rightarrow \mathbb{A}\}, \{V_p \mid p \in \mathbb{P}\} \rangle$, where W is a non-empty set of possible worlds; Each $R_i \subseteq W \times W$ is the indistinguishability relation of the agent i ; Each $T_\delta \subseteq W \times W$ is the transition relation of the joint action δ ; Each $V_p \subseteq W$ is a valuation function for p . In addition, we require that these models satisfy some constraints, for instance to ensure that it grasps correctly the concepts of knowledge and actions.

³ Expressing causality as ‘watering causes the plant to be alive’ versus ‘no watering causes plant’s death’ is more in line with the legal reasoning. In Japanese criminal law, at least in negligence cases, the relevant question is what is the duty of care to avoid damage, and the decision is whether the person violates the duty or not. In this sense, causality which mentions how to avoid damage would be more appropriate.

The satisfaction relation is the usual one for the classical connectors plus:

$$M, w \models K_G \varphi \quad \text{iff} \quad \text{for all } w' \in \bigcap_{i \in \mathbb{N}} R_i(w) \text{ we have } M, w' \models \varphi$$

$$M, w \models [\delta|_G] \varphi \quad \text{iff} \quad \text{for all } \delta' \text{ and all } w' \in T_{\delta|_G \cup \delta'|_{\mathbb{N} \setminus G}}(w) \text{ we have } M, w' \models \varphi$$

To be able to defined responsibility, we need some operators which are defined via abbreviations.

Ensuring

The formula $E_{\delta|_G} \varphi$ means ‘by executing $\delta|_G$, the group G ensures that φ ’. This operator is defined as an abbreviation:

$$E_{\delta|_G} \varphi \stackrel{\text{def}}{=} \neg[\delta|_G] \perp \wedge [\delta|_G] \varphi$$

In other words, the action δ is executable and every possible execution of it by G leads to a state where φ is true.

Ability

The formula $\langle\langle G \rangle\rangle \varphi$ means ‘group G is able to ensure φ ’. This is defined as:

$$\langle\langle G \rangle\rangle \varphi \stackrel{\text{def}}{=} \bigvee_{\delta} E_{\delta|_G} \varphi$$

In other words, there is an executable action δ such that its execution by G leads to a state where φ is true. (Note that the set of all joint actions δ is finite.)

Knowing how ability

The formula $H_G \varphi$ means ‘the group G knows how to ensure φ ’. This defined as follows:

$$H_G \varphi \stackrel{\text{def}}{=} \bigvee_{\delta} K_G E_{\delta|_G} \varphi$$

Obligations

To be able to express obligations, we add a set \mathbb{V} of violations to the logic. This set contains variables vio_G meaning ‘violation for the group G ’. Then, the formula $O_G \varphi$ means ‘it is obligatory for the group G that φ is true’, which is defined as:

$$O_G \varphi \stackrel{\text{def}}{=} \neg \varphi \rightarrow vio_G$$

Knowing causality

The formula $C_{\delta|_G} \varphi$ means ‘the group G knows that the execution of $\delta|_G$ causes φ ’. It is defined by:

$$C_{\delta|_G} \varphi \stackrel{\text{def}}{=} K_G E_{\delta|_G} \varphi \wedge \neg \langle\langle \emptyset \rangle\rangle \varphi$$

If we follow all the definitions above, we find that knowing causality amounts to group G knows that action δ is executable and its execution always lead to a state where φ is true and, in addition, it is not the case that φ is inexorably true in the next state.

Responsibility

Forward-looking responsibility is also defined as an abbreviation:

$$R_G\varphi \stackrel{\text{def}}{=} O_G H_G \varphi \wedge \langle\langle\emptyset\rangle\rangle O_G \varphi$$

Forward-looking responsibility is then defined as the obligation to have the ability to ensure φ plus the obligation that φ is true in the next state.

The definition of backward-looking responsibility, also called blame, given in [3] is based on the operators C and R. That definition can be considered a “prudent” one. Indeed, agents are blamed for φ when they knowingly cause φ . This is not enough if one wants to blame agents for outcomes that result from negligence (such as in Example 5, page 94). In this case, the following definition may be more appropriate:

$$B_{\delta|_G}\varphi \stackrel{\text{def}}{=} \neg C_{\delta|_G} \neg\varphi \wedge R_G \neg\varphi$$

In this definition, group G is blamed for φ if and only if G does not avoid the undesired outcome φ but had the forward-looking responsibility to avoid it.

Now, let us finally model Example 1 in this logic. We need one propositional variable, one agent and two actions. Let $\mathbb{P} = \{d\}$, where d means “the plant is dead”. In addition, let $\mathbb{N} = \{1\}$ and let $\mathbb{A} = \{nop, water\}$. The set of joint actions contains:

$$\alpha = \{(1, nop)\}$$

$$\beta = \{(1, water)\}$$

Now, assume a model satisfying the following formulas:

$$K_1(\neg[\alpha]\perp \wedge [\alpha]d)$$

$$K_1(\neg[\beta]\perp \wedge [\beta]\neg d)$$

$$R_1\neg d$$

The first formula means ‘agent 1 knows that α is executable and its execution leads to a state where the plant is dead’. The meaning of second one is similar. The third formula means ‘1 is forward-looking responsible for the plant is not dead’.

Because there is an action after which the plant is not dead, the model satisfies $\neg\langle\langle\emptyset\rangle\rangle d$. Then, the model also satisfies $C_{\alpha|_1} d$, which implies $\neg C_{\alpha|_1} \neg d$. This means that the model satisfies $B_{\alpha|_1} d$. In other words, agent 1 is blamed for d .

4.1.2.3 Causal models [6, 2, 1]

In this section, we consider the approach to formalising responsibility proposed by Chockler and Halpern [2]. We first briefly review Halpern’s definition of causality [6] and Chockler and Halpern’s definition of responsibility and blame [2]. Much of the description below is taken from [6]. The Halpern and Pearl approach (hereafter HP) assumes that the world is described in terms of variables and their values. Some variables may have a causal influence on others. This influence is modelled by a set of *modifiable structural equations*. Variables are split into two sets: the *exogenous* variables, whose values are determined by factors outside the model, and the *endogenous* variables, whose values are ultimately determined by the exogenous variables. The structural equations describe how the outcome is determined.

Formally, a *causal model* M is a pair $(\mathcal{S}, \mathcal{F})$, where \mathcal{S} is a *signature* and \mathcal{F} is a function that associates a structural equation with each variable. A signature \mathcal{S} is a tuple $(\mathcal{U}, \mathcal{V}, \mathcal{R})$,

where \mathcal{U} is a set of exogenous variables, \mathcal{V} is a set of endogenous variables, and \mathcal{R} associates with every variable $Y \in \mathcal{U} \cup \mathcal{V}$ a nonempty set $\mathcal{R}(Y)$ of possible values for Y (i.e., the set of values over which Y ranges). \mathcal{F} associates with each endogenous variable $X \in \mathcal{V}$ a function denoted F_X such that $F_X : (\times_{U \in \mathcal{U}} \mathcal{R}(U)) \times (\times_{Y \in \mathcal{V} - \{X\}} \mathcal{R}(Y)) \rightarrow \mathcal{R}(X)$. Thus, F_X defines a structural equation that determines the value of X given the values of other variables. Setting the value of some variable X to x in a causal model $M = (\mathcal{S}, \mathcal{F})$ results in a new causal model, denoted $M_{X \leftarrow x}$, which is identical to M , except that the equation for X in \mathcal{F} is replaced by $X = x$.

Given a signature $\mathcal{S} = (\mathcal{U}, \mathcal{V}, \mathcal{R})$, a *primitive event* is a formula of the form $X = x$, for $X \in \mathcal{V}$ and $x \in \mathcal{R}(X)$. A *causal formula (over \mathcal{S})* is one of the form $[Y_1 \leftarrow y_1, \dots, Y_k \leftarrow y_k] \varphi$, where φ is a Boolean combination of primitive events, Such a formula is abbreviated as $[\vec{Y} \leftarrow \vec{y}] \varphi$. The special case where $k = 0$ is abbreviated as φ . Intuitively, $[Y_1 \leftarrow y_1, \dots, Y_k \leftarrow y_k] \varphi$ says that φ would hold if Y_i were set to y_i , for $i = 1, \dots, k$.

Following [6, 8], we only consider *acyclic* models. In acyclic models, there is a total ordering \prec of the endogenous variables such that if $X \prec Y$, then X is independent of Y , that is, $F_X(\vec{z}, y, \vec{v}) = F_X(\vec{z}, y', \vec{v})$ for all $y, y' \in \mathcal{R}(Y)$. If $X \prec Y$, then the value of X may affect the value of Y , but the value of Y cannot affect the value of X . If M is an acyclic causal model, then given a *context*, that is, a setting \vec{u} for the exogenous variables in \mathcal{U} , there is a unique solution for all the equations: it is possible to solve the equations for the variables in the order given by \prec . A causal formula ψ is true or false in a causal model, given a context. We write $(M, \vec{u}) \models \psi$ if the causal formula ψ is true in causal model M given context \vec{u} . The \models relation is defined inductively. $(M, \vec{u}) \models X = x$ if the variable X has value x in the unique (since we are dealing with acyclic models) solution to the equations in M in context \vec{u} . The truth of conjunctions and negations is defined in the standard way. Finally, $(M, \vec{u}) \models [\vec{Y} \leftarrow \vec{y}] \varphi$ if $(M_{\vec{Y}=\vec{y}}, \vec{u}) \models \varphi$. Thus, $[\vec{Y} \leftarrow \vec{y}] \varphi$ is true in (M, \vec{u}) if φ is true in the model that results after setting the variables in \vec{Y} to \vec{y} .

The causal model M_1 for Example 1 is as follows (note that we introduce the variable for a normative fact of an obligation in addition to the plain facts):

- $\mathcal{U}_1 = \{A_1\}$ is the set of exogenous variables; A_1 corresponds to Agent 1's intention of watering the plant;
- $\mathcal{V}_1 = \{ObF, D, W\}$ is the set of endogenous variables; ObF stands for obligation fulfilled, D for the plant is dead, W for the agent waters the plant
- \mathcal{R} is given by the following structural equations:
 - $W = A_1$;
 - $ObF = W$;
 - $D = \neg W$;

The context is $\{\neg A_1\}$.

Next we define causality. Causality is relative to a model and a context. Only conjunctions of primitive events, abbreviated as $\vec{X} = \vec{x}$, can be causes. What can be caused are arbitrary Boolean combinations of primitive events. Roughly speaking, $\vec{X} = \vec{x}$ is a cause of φ if, had $\vec{X} = \vec{x}$ not been the case, φ would not have happened. To deal with many well-known examples (see [6]), the actual definition is more complicated.

► **Definition 2.** $\vec{X} = \vec{x}$ is an *actual cause* of φ in (M, \vec{u}) if the following three conditions hold:

AC1. $(M, \vec{u}) \models (\vec{X} = \vec{x})$ and $(M, \vec{u}) \models \varphi$.

AC2^m. There is a set \vec{W} of variables in \mathcal{V} and settings \vec{x}' of the variables in \vec{X} and \vec{w} of the variables in \vec{W} such that $(M, \vec{u}) \models \vec{W} = \vec{w}$ and

$$(M, \vec{u}) \models [\vec{X} \leftarrow \vec{x}', \vec{W} \leftarrow \vec{w}] \neg \varphi.$$

AC3. \vec{X} is minimal; no subset of \vec{X} satisfies conditions AC1 and AC2^m.

AC1 states that for $\vec{X} = \vec{x}$ to be a cause of φ , both $\vec{X} = \vec{x}$ and φ have to be true. AC3 is a minimality condition, which ensures that only the conjuncts of $\vec{X} = \vec{x}$ that are essential are parts of a cause. AC2^m (the “m” is for modified; the notation is taken from [6]) captures the counterfactual. It says that if we change the value of \vec{X} from \vec{x} to \vec{x}' , while possibly holding the values of the variables in some (possibly empty) set \vec{W} fixed at their values in the current context, then φ becomes false. We say that (\vec{W}, \vec{x}') is a *witness* to $\vec{X} = \vec{x}$ being a cause of φ in (M, \vec{u}) . If $\vec{X} = \vec{x}$ is a cause of φ in (M, \vec{u}) and $X = x$ is a conjunct of $\vec{X} = \vec{x}$, then $X = x$ is *part of a cause* of φ in (M, \vec{u}) .

In Example 1, the cause of $\neg ObF$ is $\neg W$, and the cause of D is also $\neg W$. The witness is empty in both cases.

The notion of *degree of responsibility* was introduced by Chockler and Halpern in [2]. Roughly speaking, the degree of responsibility $X = x$ for φ measures the minimal number of changes and number of variables that have to be held fixed in order to make φ counterfactually depend on $X = x$. We use the formal definition in [7], which is appropriate for the modified definition of causality used here.

► **Definition 3.** The *degree of responsibility* of $X = x$ for φ in (M, \vec{u}) , denoted $dr((M, \vec{u}), (X = x), \varphi)$, is 0 if $X = x$ is not part of a cause of φ in (M, \vec{u}) ; it is $1/k$ if there exists a cause $\vec{X} = \vec{x}$ of φ and a witness (\vec{W}, \vec{x}') to $\vec{X} = \vec{x}$ being a cause of φ in (M, \vec{u}) such that (a) $X = x$ is a conjunct of $\vec{X} = \vec{x}$, (b) $|\vec{W}| + |\vec{X}| = k$, and (c) k is minimal, in that there is no cause $\vec{X}_1 = \vec{x}_1$ for φ in (M, \vec{u}) and witness (\vec{W}', \vec{x}'_1) to $\vec{X}_1 = \vec{x}_1$ being a cause of φ in (M, \vec{u}) that includes $X = x$ as a conjunct with $|\vec{W}'| + |\vec{X}_1| < k$.

In Example 1, the degree of responsibility of $\neg W$ (essentially, agent 1’s (in) action), for both $\neg ObF$ and D is 1: $dr((M_1, \{\neg A_1\}), (\neg W), \neg ObF) = 1$.

This definition of responsibility assumes that everything relevant about the facts of the world and how the world works is known. In general, there may be uncertainty about both. The notion of *blame* takes this into account. We model an agent’s uncertainty by a pair (\mathcal{K}, Pr) , where \mathcal{K} is a set of causal settings, that is, pairs of the form (M, \vec{u}) , and Pr is a probability distribution over \mathcal{K} . We call such a pair an *epistemic state*. Note that once we have such a distribution, we can talk about the probability that $\vec{X} = \vec{x}$ is a cause of φ relative to (\mathcal{K}, Pr) : it is just the probability of the set of pairs (M, \vec{u}) such that $\vec{X} = \vec{x}$ is a cause of φ in (M, \vec{u}) . We also define the *degree of blame* of $X = x$ for φ to be the expected degree of responsibility:

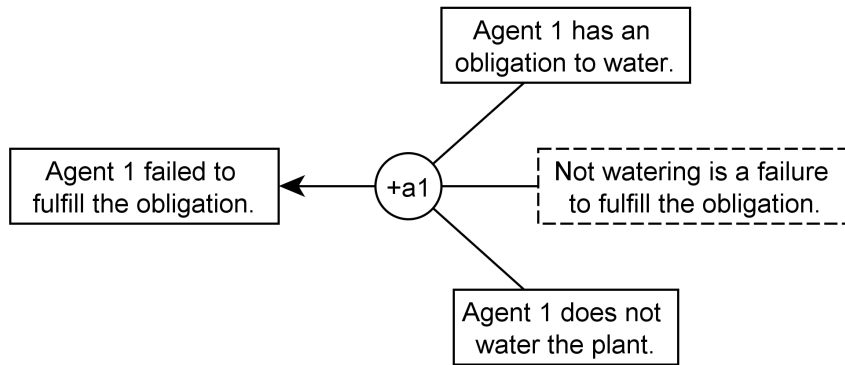
► **Definition 4.** The *degree of blame* of $X = x$ for φ relative to the epistemic state (\mathcal{K}, Pr) is

$$\sum_{(M, \vec{u}) \in \mathcal{K}} dr((M, \vec{u}), X = x, \varphi) \text{Pr}((M, \vec{u})).$$

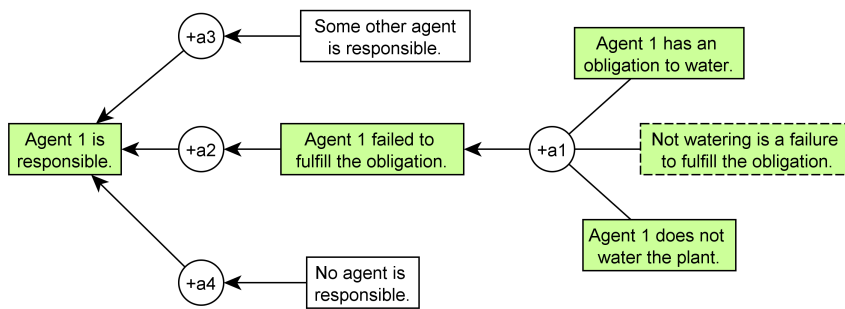
In Example 1, the degree of blame of $\neg W$ may be quite low if $\text{Pr}((M_1, \{\neg A_1\}))$ is low; for example, the agent may not know the structural equation for ObF and assign probability 0 to M_1 .

4.1.2.4 Argumentation theory

The Carneades Argumentation System, named after the Greek skeptical philosopher, is open source software, available at <http://carneades.github.io/>. It is a computational system, because the model consists of mathematical structure whose operations are all computable.



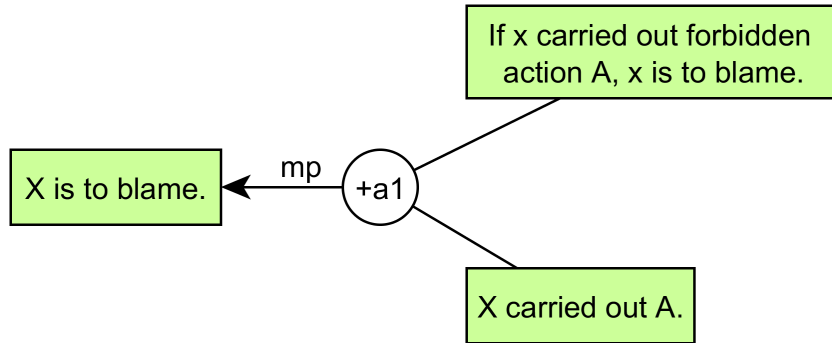
■ **Figure 1** Example 1 propositions and arguments.



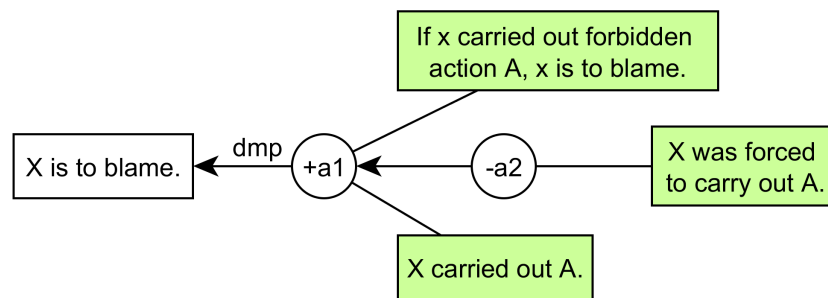
■ **Figure 2** Example 1 with responsibility accepted.

It is also a formal system. Carneades formalises argument graphs, as bipartite, directed graphs, consisting of argument nodes linked to statement nodes. Argument graphs model inferential relationships among arguments and statements. An argument graph is a bipartite, directed, labeled graph, consisting of statement nodes and argument nodes connected by premise and conclusion edges. Formally, an argument graph is a 4-tuple $\langle S, A, P, C \rangle$, where S is a set of statement nodes, A is a set of argument nodes, P is a set of premises, and C is a set of conclusions. To see examples, look ahead to Figures 1-5.

The argument diagrams shown below are graph structures drawn in the style of the Carneades Argumentation System, see, for example, [15]. The sentences in the rectangular nodes denote propositions. The circular nodes represent arguments, which can be pro or con a proposition, or an argument. The propositions are premises or conclusions of arguments. Several premises can support the conclusion together in what is called the linked argument configuration. Or two or more propositions can independently support a conclusion in what is called convergent argumentation structure in informal logic. Implicit premises or conclusions are indicated by the dashed perimeter of the rectangular node. The general idea is that arguments have a graph structure, and in the most typical instances there is an ultimate conclusion to be proved or disproved that is represented as a root of an argumentation tree. By this means the sequence of argumentation on both sides of a disputed issue can be visually represented, and in the end the pro-arguments can be weighed against con arguments so that it can be charged which side had the stronger argument using standards and burdens of proof. An example is shown in Figure 1.



■ **Figure 3** Deductive Modus Ponens Argument.



■ **Figure 4** Pollock-Style Undercutter.

When a rectangular node has a green background, it means that this proposition has been accepted by the audience (in many instances, the user). Based on the user's input, Carneades can use argumentation schemes to calculate whether a conclusion is accepted (labelled 'in') based on a set of premises. An example is shown in Figure 2.

One might initially tend to think that the problem of assigning blame to an agent in a normative multiagent system can simply be dealt with by applying the following rule (R1): if agent x carries out action A , and action A is forbidden in the normative system, then x is to blame for carrying out A . And in general R1 might work as a base principle for assigning blame in a normative system, but there are two problems with applying it to real cases.

The first problem, the defeasibility of this principle, was extensively discussed long ago by [9], and his contemporaries. Suppose, for example, that x did carry out action A , but this action was not voluntary, because it fell under the category of one of a list of defeating conditions. For example, suppose x was forced to carry out action A by another agent y . In such an instance it may be true that agent x carried out action A , and that action A is forbidden in the normative system, but it might not be true that x is to blame for carrying out action A . As Hart pointed out, there might be a long, even open-ended list of such defeating conditions.

How this problem is modeled in formal argumentation systems can be seen by considering the kind of structure pictured in Figure 3. The plus sign represents a pro argument, meaning that the premises are put forward to support acceptance of the conclusion.

The argument in this instance is based on the rule of *modus ponens* as standardly defined in classical deductive logic. If both premises are true, it follows deductively that the conclusion

has to be true. In Figure 3 both premises are colored in green, indicating that both have been accepted by the audience. This means that in a formal argumentation system, such as Carneades, the system will automatically color the conclusion in green.

However, let's go on to consider what happens if we model the inference not by using the deductive version of *modus ponens*, but by defeasible *modus ponens*. When a defeasible rule of inference is used, the acceptance of both premises shifts a weight of presumption towards acceptance of the conclusion, but does not require that the conclusion has to be true. To see how this kind of inference rule works, we need to consider some different ways of attacking and defeating an argument characteristic of argumentation theory.

It is widely recognized in formal argumentation systems of the kind studied in artificial intelligence that there are three ways of attacking an argument: you can attack one or more of the premises, you can attack the conclusion, or you can attack the inferential link joining the premises to the conclusion [11]. The third way is associated with the form of argument attack called a Pollock-style undercutter, which can be illustrated by Pollock's [10] classic example. Suppose I am looking at a light, and it looks red to me, but I also know that it is illuminated by a red light and that red lights can make an object look red even when they are not. Note that the new evidence does not rebut the claim that the object is red, because it might be red for all I know. It merely undercuts the original argument, meaning that it casts the original argument into doubt by undermining the rule that anything that looks red has to be red [14].

This same kind of reasoning can be applied to reasoning from a forbidden action to blame.

By looking at Figure 4, we can see how the defeasible argumentation applies to drawing a conclusion that an agent is to blame for a particular action based on the premises that this action was forbidden and that the agent carried out. Both premises are accepted, and hence in Figure 4 are shown in green, just as they were in Figure 3. But in Figure 4 the inference to the conclusion is based on defeasible *modus ponens* (dmp), which leaves the inference to the conclusion open to being undercut by new information that has come into a particular case [14]. In this instance, the new information is that the agent was forced to carry out the action in question. This finding acts as a con argument, shown as argument a2 in Figure 4, where the minus sign indicates a con argument, an argument that has been put forward to attack a prior argument.

The problem posed by these considerations can be addressed by an argumentation system which allows some arguments to attack other arguments, and in particular which allows for the use of defeasible forms of argument such as defeasible *modus ponens*. However, portraying an inference from premises about an agent's action, and whether these actions are forbidden, to a conclusion that the agent was to blame as a deductive form of argument, does not take defeasibility into account. This is a severe limitation in studying how ethical and legal reasoning are accounted for when studying how to reason properly about responsibility.

The second problem is that the action A that x carried out might have set a chain of consequences into motion, and one of these consequences might constitute an outcome that is forbidden to bring about in the normative system. In such a case, x might correctly have been seen to be properly blamed for carrying out action A, even though A in itself was not forbidden in the normative system. This takes us to the task of modeling the indirect consequences of an agent's actions through causal sequences in order to show how to reason properly about responsibility in multiagent systems.

4.1.3 Distinguishing between causality and responsibility

In this section, we consider a more complex case, in which an agent is obliged to perform an action, and more than one agent may act. In this setting, responsibility for a state of affairs, and responsibility for violation of the norm do not necessarily coincide.

► **Example 5.** As before, a plant must be watered in order to prevent it dying. Agent 1 has an obligation to water the plant. Agent 1 does not water the plant. Agent 2 could have watered the plant (is able to see that the plant is not watered, and is able to water it) but didn't. The plant dies. Who is responsible for the death of the plant? Who is responsible for the violation of the obligation?

4.1.3.1 Modal logic with three modalities

The existence of Agent 2 does not change the analysis for this example. Agent 1 is still responsible for violating the obligation and the dead plant. Agent 2 did not have an obligation to water the plant, and hence is not responsible.

4.1.3.2 Logic of strategic ability

Similarly as before, we can model Example 5 with $\mathbb{P} = \{d\}$, $\mathbb{N} = \{1, 2\}$ and $\mathbb{A} = \{nop, water\}$, and also:

$$\begin{aligned}\alpha &= \{(1, nop), (2, nop)\} \\ \beta &= \{(1, nop), (2, water)\} \\ \gamma &= \{(1, water), (2, nop)\} \\ \delta &= \{(1, water), (2, water)\}\end{aligned}$$

Assume a model satisfying:

$$\begin{aligned}\mathbf{K}_{\mathbb{N}}(\neg[\alpha] \perp \wedge [\alpha]d) \\ \mathbf{K}_{\mathbb{N}}(\neg[\beta] \perp \wedge [\beta] \neg d) \\ \mathbf{K}_{\mathbb{N}}(\neg[\gamma] \perp \wedge [\gamma] \neg d) \\ \mathbf{K}_{\mathbb{N}}(\neg[\delta] \perp \wedge [\delta] \neg d) \\ \mathbf{R}_1 \neg d \\ \neg \mathbf{R}_2 \neg d\end{aligned}$$

We have that the model satisfies $\neg C_{\alpha|_1} \neg d$. (Also note that $\alpha|_1$ is the same as $\beta|_1$.) This means that it also satisfies $B_{\alpha|_1} d$. In other words, agent 1 is blamed for the death of the plant. However, since agent 2 did not have forward-looking responsibility for the plant, 2 is not blamed for the undesirable outcome.

4.1.3.3 Causal models

The model M_2 for Example 5 is as follows:

- $\mathcal{U}_2 = \{A_1, A_2\}$ is the set of exogenous variables; A_i corresponds to Agent i 's intention of watering the plant;
- $\mathcal{V}_2 = \{ObF, D, W_1, W_2\}$ is the set of endogenous variables; ObF stands for obligation fulfilled, D for the plant is dead, W_i for agent i waters the plant;
- \mathcal{R} is given by the following structural equations:
 - $W_1 = A_1$;

- $W_2 = A_2$;
- $ObF = W_1$;
- $D = \neg W_1 \wedge \neg W_2$;

The context is $\{\neg A_1, \neg A_2\}$.

Now the cause of $\neg ObF$ is still $\neg W_1$, but the cause of D is $\neg W_1, \neg W_2$. That is, as in the Strategic Ability model, we can distinguish between the cause of the plant's death, and responsibility for the failure to fulfil the obligation.

4.1.3.4 Argumentation theory

In this section, we consider how argumentation theory can be used to give an explanation of some aspects of the reasoning in Example 5. We begin by changing the question asked in the example according to the following formulation, with the aim of trying to understand what general ethical principle could be applied to the specific circumstances of the case.

In Example 5, Agent 1 has an obligation to water the plant. Agent 1 did not water the plant. But agent 1 knew that if he did not water the plant the plant will die. The plant dies. Agent 2 has no obligation to water the plant. But agent 2 also knew that if she did not water the plant the plant will die. Who is to blame for the death of the plant?

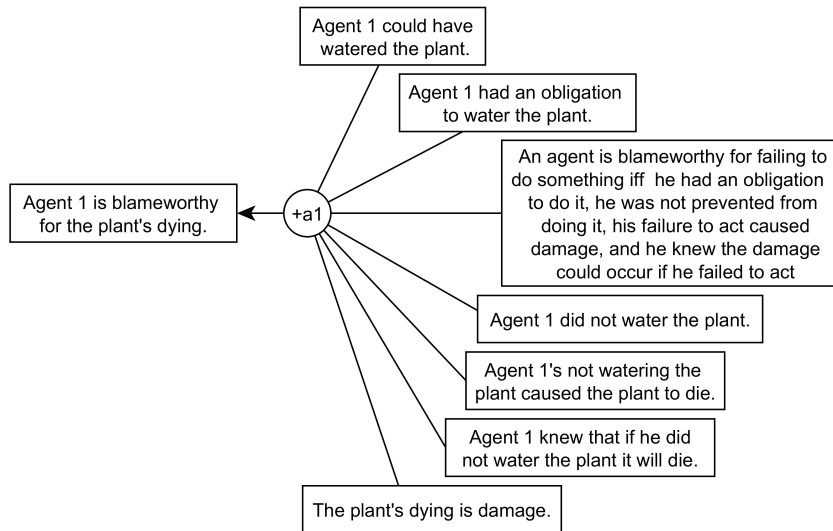
From the text of this example, the argumentation expressed in it can be represented as an instance of case-based reasoning based on an implicit ethical generalization stating a set of conditions that are necessary and sufficient for drawing the conclusion that agent 1 is blameworthy for the plant's dying. The generalization is the statement that an agent is blameworthy for failing to do something if and only if the agent had an obligation to do it, he was not prevented from doing it, his failure to act caused damage, and he knew the damage could occur if he failed to act. Each one of the four conditions is taken to be necessary in the generalization to support the inference to the conclusion that agent one is blameworthy for the plant's dying, and the conjunction of the four conditions is taken to be sufficient to support the conclusion that agent one is blameworthy for the plant's dying. The argumentation structure representing this reasoning is shown in Figure 5.

What this means in the Carneades Argumentation System is that if all seven premises of argument a1 are accepted, the conclusion of argument a1 should be accepted as following from them.

But what about the case of agent 2? The argument diagram for the case of agent 2 is the same as the case of agent 1, as shown in Figure 5, except that the second premise from the top, containing the proposition that agent 1 had an obligation to water the plant, does not hold. This means that even although the other six premises do hold, and are therefore colored green in the diagram, the conclusion now fails to hold. So the conclusion is colored with a white background, showing that it is no longer accepted, based on the argument.

One conclusion that can be drawn from this way of structuring the argumentation in the case is that the generalization shown in the large rectangular node in figure 5, once combined in an argument structure with the other premises specifying the factual circumstances taken to hold in the case, provides sufficient support for us to draw the conclusion that agent 1 is blameworthy. Another conclusion that can be drawn is that once the premise that agent 2 had an obligation to water the plant is no longer accepted as holding in this variant of the case, the conclusion that agent 2 is blameworthy is no longer supported as acceptable.

The question now raised is whether the ethical generalization used to draw the conclusions about blameworthiness based on the differing circumstances of the two agents is the correct basis for drawing this conclusion. In other words, is this generalization the correct ethical principle that should generally be used for deciding whether or not an agent is blameworthy



■ **Figure 5** Argument for the Responsibility of Agent 1 in the Plant Example.

when an agent has carried out actions fitting the requirements of the circumstances specified in the two examples? So far, it can stand as a hypothesis that this ethical principle can provide a provisional basis for drawing conclusions of this sort in specific cases. It can be put in place as a starting point for investigating further more complex cases where the circumstances are varied to fit problematic cases of assigning blame and responsibility. If or when counter-examples are found in the new cases, the generalization may have to be modified or even given up.

4.1.4 Unintentional violation

► **Example 6.** Agent 1 has an obligation to water the plant. Agent 1 does not water the plant because it is raining and agent 1 assumes the plant will get watered by the rain. However the plant is under cover and does not get watered by the rain. Agent 2 could have watered the plant (is able to see that the plant is not watered, and is able to water it) but didn't. The plant dies. Who is responsible for the death of the plant? Who is responsible for the violation of the obligation?

4.1.4.1 Modal logic with three modalities

This is the case of error/negligence on the part of agent 1: $\neg K_1(w_1 \rightarrow \neg d)$ (or, $\neg K_1(\neg w_1 \rightarrow d)$).

4.1.4.2 Logic of strategic ability

We cannot model this example correctly in CEDL. The reason is that, in the example, the agent “believes” (instead of “knows”) that the rain will water the plant. The notion of belief is different than that of knowledge. If an agent believes φ then φ is true in all situations that are considered possible by the agent but (as in Example 6) the agent may be wrong. In terms of Kripke semantics, this means that the actual situation may not be one of the

situations that the agent considers possible. Even more technically, this means that the axiom T ($K\varphi \rightarrow \varphi$) is not valid in a logic modelling beliefs. However, it is valid for knowledge, and thus, valid in CEDL.

This is why CEDL, as it stands, cannot model the problem. To better understand it, let the formula $K_i[\alpha]\neg d$ stand for ‘the agent knows that, after the rain (α), the plant is not dead’. By axiom T, we must have $[\alpha]\neg d$, which means ‘after the rain, the plant is not dead’. The latter cannot be the case in Example 6.

To avoid the latter problem, one may propose to just replace axiom T by axiom D and thus work with a logic where operator K means belief instead of knowledge. The operator K in this case would be the common operator for beliefs, for example, studied in [5]. But there is another axiom in this logic that may cause problems. The so-called ‘no-forgetting’ principle, which is as follows:

$$K_G[\alpha|_G]\varphi \rightarrow [\alpha|_G]K_G\varphi$$

This principle implies that the knowledge (or in this case the beliefs) of agents either increase or stay the same after the execution of any action. The presence of ‘no-forgetting’ prevents situations where agents come to know (or believe) something that contradicts what was known (or believed) before. If we get back to our example, we have that, at first, the agent thinks that the plant will be alive after the rain, but once the plant dies, the agent still thinks that it is alive. For instance, let the formula $K_i[\alpha|_{\mathbb{N}}]\neg d$ stand for ‘the agent believes that, after the rain (α) the plant is not dead’. By ‘no-forgetting’, we must have $[\alpha|_{\mathbb{N}}]K_i \neg d$, which means ‘after the rain, the agent believes that the plant is not dead’. The result is a logic where, if the agent believes something that is not correct, the agent will keep believing it, no matter what.

We may, nonetheless, try to model Example 6 in this new formalism. First, we have to add a third agent that represents the environment. (This can be seen as “the rain” in the example.) The set of agents is thus $\mathbb{N} = \{1, 2, 3\}$. Every agent, including the environment agent 3, may water the plant or not. Hence, the sets of actions for each agent are $\mathbb{A}_i = \{water, skip\}$, for $i \in \mathbb{N}$. The joint actions are thus all the combinations of these two actions: $\{(1, skip), (2, skip), (3, skip)\}, \{(1, skip), (2, skip), (3, water)\}, \dots$. Now, since agent 1 thinks that the rain will water the plant, we could try to assume a model satisfying the following formula:

$$\begin{aligned} &K_1[(3, skip)]\perp \\ &K_1(\neg[(3, water)]\perp \wedge [(3, water)]\neg d) \end{aligned}$$

These formulas mean that agent 1 believes that agent 3 cannot skip and hence agent 3 necessarily waters the plant. However, by ‘no-forgetting’, we must have:

$$[(3, skip)]K_1\perp$$

The latter is inconsistent with axiom D. To try to find a way around this problem, we may consider that the agent believes that action skip also waters the plant. In this case we have, instead, a model satisfying:

$$\begin{aligned} &K_1([(3, skip)]\neg\perp \wedge [(3, skip)]\neg d) \\ &[(1, skip), (2, skip), (3, skip)]\neg\perp \wedge [(1, skip), (2, skip), (3, skip)]d \end{aligned}$$

These two formulas mean that agent 1 believes that, after agent 3 skips, the plant is alive, but actually this is not the case. For the other actions, we have the usual. Thus assume that

the model also satisfies:

$$\begin{aligned}
& K_{\mathbb{N}}(\neg[(1, skip), (2, skip), (3, water)] \perp \wedge [(1, skip), (2, skip), (3, water)] \neg d) \\
& K_{\mathbb{N}}(\neg[(1, skip), (2, water), (3, skip)] \perp \wedge [(1, skip), (2, water), (3, skip)] \neg d) \\
& \vdots \\
& R_1 \neg d \\
& \neg R_2 \neg d \\
& \neg R_3 \neg d
\end{aligned}$$

Because there is an action after which the plant is dead, the model satisfies $\neg\langle\langle\emptyset\rangle\rangle\neg d$. We also have that agent 1 believes that skipping ensures that the plant will be alive: $K_1 E_{(1, skip)} \neg d$. Then, by the definitions given, the agent “believably” causes that the plant is alive after skipping: $C_{\alpha_1} \neg d$ (even though it may actually be dead). This means that the model satisfies $\neg B_{(1, skip)} d$. In other words, agent 1 is not blamed for the eventual death of the plant. The reason that agent 1 is excused is that the agent believes that the death of the plant is prevented. One may of course wonder whether this is enough to excuse the agent.

4.1.4.3 Causal models

Responsibility stays the same (agents are both causally responsible). However under the reasonable probability distribution over possible models (where the chance of the plant not being watered by the rain when it is raining is very small) the degree of blame attached to agent 1 is small.

4.1.4.4 Argumentation theory

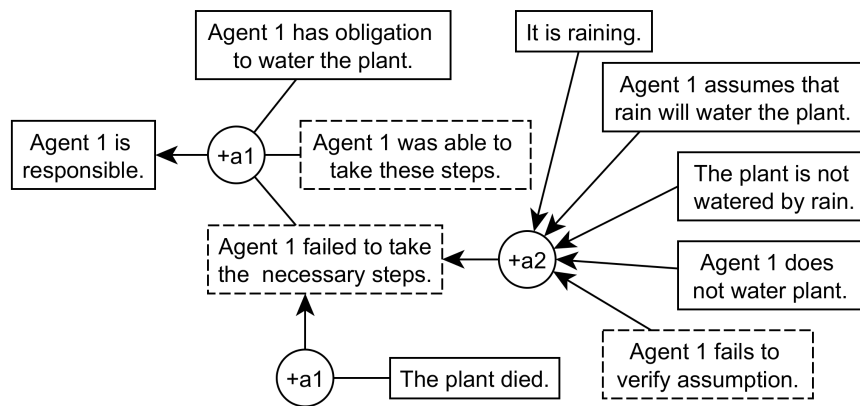
In example 6 two agents are involved. One of them wrongly assumes that the plant will get watered by rain, but this does not turn out to be true. The other agent could have also watered the plant but didn’t, and so the plant dies. The arguments in this case are composed from seven propositions stated in the key list below.

Key List for Responsibility of Agent 1:

- (1) Agent 1 has an obligation to water the plant.
- (2) Agent 1 does not water the plant.
- (3) It is raining.
- (4) Agent 1 assumes that the plant will get watered by the rain.
- (5) The plant does not get watered by the rain.
- (6) The plant died.
- (7) Agent 1 is responsible for the death of the plant.

It is known that the plant did not get watered by the rain because it was under cover. This is an explanation of why the plant did not get watered (as opposed to an argument), so it was not included in the argument diagram in Figure 6. However, three implicit premises need to be inserted in order for us to make sense of the argumentation in the example. The following three implicit premises are shown in rectangles with broken-line perimeters.

- (8) Agent 1 failed to verify his assumption that the plant will get watered by the rain.
- (9) Agent 1 failed to take the necessary steps to see to it that the plant was watered.
- (10) Agent 1 was able to take these necessary steps.



■ **Figure 6** Argument Diagram for Agent 1 in Example 6.

Here is the general ethical principle behind drawing inferences about responsibility and blame in cases concerning unintentional violations of an obligation. An agent can be held responsible for failing to fulfill an obligation even if he thought it would be fulfilled in the normal course of events without his taking any further steps to see that this happens. This can occur where the agent had some reason to assume that in the circumstances he does not need to intervene to see to it that the obligation is fulfilled. If it was not, he can be held responsible for failing to fulfill his obligation on the grounds that he failed to take steps that he could have and should have taken to see to it that the bad outcome he was obliged to prevent from occurring did not occur. In law this sort of principle applies to judging cases of responsibility relating to failures such as ‘taking due care’ or taking precautions. Next we need to consider the responsibility of agent 2. Here we have a key list of five propositions and we need to add one implicit premise.

Key List for Responsibility of Agent 2

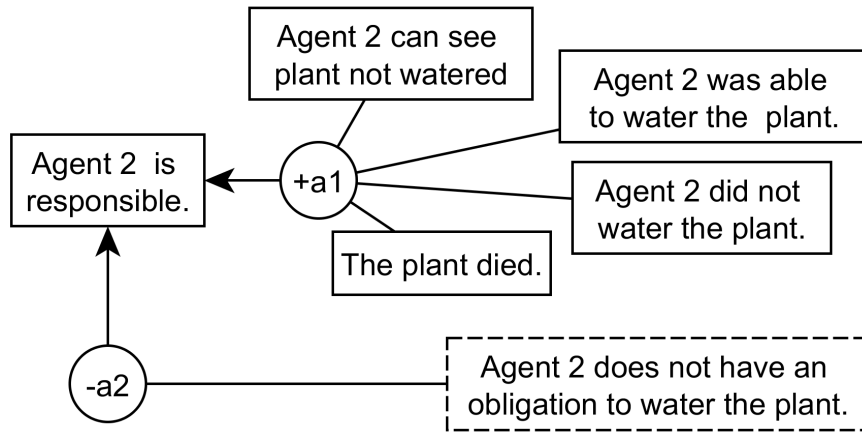
- (1) Agent 2 is able to see that the plant is not watered.
- (2) Agent 2 is able to water the plant.
- (3) Agent 2 did not water the plant.
- (4) The plant died.
- (5) Agent 2 is responsible for the death of the plant.

Implicit Premise

- (6) Agent 2 does not have an obligation to water the plant.

Based on this interpretation of the argument in Example 6 concerning the responsibility of agent 2, the argument diagram shown in Figure 8 shows that there is a pro argument +a1 supporting the conclusion that agent 2 is responsible, but there is also a con argument –a2 attacking the conclusion that agent 2 is responsible.

The con argument a2 shows that the pro argument is not strong enough to prove its conclusion because, as shown in the ethical principle formulated in connection with Example 5 (see Figure 5), having an obligation to carry out action is a necessary requirement to draw the conclusion that an agent can be held responsible for failure to carry out the action. Hence in this instance, the con argument defeats the pro argument.



■ **Figure 7** Argument Diagram for Agent 2 in Example 6.

4.1.5 Causality revisited (and knowledge of strategy)

► **Example 7.** Agents 1 and 2 have an obligation that exactly one of them waters the plant (if both of them do, the plant also dies). None of them waters the plant. The plant dies. Who is responsible for the death of the plant? Who is responsible for the violation of the obligation?

4.1.5.1 Modal logic with three modalities

Suppose both agents have an obligation of ‘not too much watering’ since otherwise the plant dies. Suppose that both agents watered too much and the plant died. Using *conditio sine qua non*, the conclusion is derived:

- even if Agent 1 had not watered too much, the plant would have died anyway so Agent 1 is not responsible and
- even if Agent 2 had not watered too much, the plant would have died anyway so Agent 2 is not responsible.

For analysis, see [13].

4.1.5.2 Logic of strategic ability

This is similar to a previous example. The only difference on the modelisation is that exactly one of the agents is responsible: Let the group of agents be $G = \{1, 2\}$, and assume a model satisfying:

$$R_G \neg d \\ (R_1 \neg d \vee R_2 \neg d) \wedge (\neg R_1 \neg d \vee \neg R_2 \neg d)$$

As before, we have that the model satisfies $C_{\alpha|G} d$. This means that $B_{\alpha|G} d$ is satisfied and thus, group G is blamed for the death of the plant.

The difference here is that exactly one of the agents is individually blamed for the death of the plant. That is, we have:

$$(B_{\alpha|1} d \vee B_{\alpha|2} d) \wedge (\neg B_{\alpha|1} d \vee \neg B_{\alpha|2} d)$$

4.1.5.3 Causal models

Each of the agents is individually responsible (if everything else stayed the same and agent 1 watered the plant, the plant would have been alive; similarly for agent 2). The degree of blame is not 1 however since there is a non-zero probability that agent 1 watering the plant (in the context where agent 2 also waters the plant) would have caused it to die.

4.1.6 Group responsibility

► **Example 8.** Agents 1 and 2 have an obligation to water the plant. Neither of them does. The plant dies. Who is responsible for the death of the plant? Who is responsible for the violation of the obligation?

In this case we consider only logic of strategic ability and causal models.

4.1.6.1 Logic of strategic ability

This example can be modeled as follows. Similarly as before we have:

$$\begin{aligned}\alpha &= \{(1, \text{nop}), (2, \text{nop})\} \\ \beta &= \{(1, \text{nop}), (2, \text{water})\} \\ \gamma &= \{(1, \text{water}), (2, \text{nop})\} \\ \delta &= \{(1, \text{water}), (2, \text{water})\}\end{aligned}$$

Let the group of agents be $G = \{1, 2\}$, and assume a model satisfying:

$$\begin{aligned}\mathsf{K}_{\mathbb{N}}(\neg[\alpha] \perp \wedge [\alpha]d) \\ \mathsf{K}_{\mathbb{N}}(\neg[\beta] \perp \wedge [\beta]\neg d) \\ \mathsf{K}_{\mathbb{N}}(\neg[\gamma] \perp \wedge [\gamma]\neg d) \\ \mathsf{K}_{\mathbb{N}}(\neg[\delta] \perp \wedge [\delta]\neg d) \\ \mathsf{R}_G \neg d\end{aligned}$$

We have that the model satisfies $C_{\alpha|G}d$. This means that it also satisfies $B_{\alpha|G}d$. In other words, group G is blamed for the death of the plant.

Note that no agent is individually held forward-looking responsible for the plant is not dead. This is why agents 1 and 2 are not blamed individually. However, if one of them, e.g. 1, is held individually responsible, i.e. $\mathsf{R}_1 \neg d$, then it would be held responsible, exactly as in the previous example. The same for agent 2.

4.1.6.2 Causal models

The cause of the plant dying is that neither agent watered the plant; so both agents' actions are part of a single cause. The degree of responsibility is therefore 1/2 for each agent. The degree of blame depends on the probability distribution; it is reasonable to assume that it is the same as the degree of responsibility (that is, the given context has probability 1).

References

- 1 Natasha Alechina, Joseph Y. Halpern, and Brian Logan. Causality, responsibility and blame in team plans. In S. Das, E. Durfee, K. Larson, and M. Winikoff, editors, *Proceedings of the 16th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2017)*, Sao Paulo, Brazil, May 2017. IFAAMAS, IFAAMAS.

- 2 H. Chockler and J. Y. Halpern. Responsibility and blame: A structural-model approach. *Journal of Artificial Intelligence Research*, 20:93–115, 2004.
- 3 Tiago de Lima and Lambèr Royakkers. A formalisation of moral responsibility and the problem of many hands. In Ibo van de Poel, Lambèr Royakkers, and Sjoerd D. Swart, editors, *Moral Responsibility and the Problem of Many Hands*, Routledge Studies in Ethics and Moral Theory, chapter 3, pages 93–130. Taylor & Francis, 2015.
- 4 Tiago de Lima, Lambèr Royakkers, and Frank Dignum. Modeling the problem of many hands in organisations. In H. Coelho, Rudi Studer, and Michael Wooldridge, editors, *Proceedings of the 19th European Conference on Artificial Intelligence (ECAI 2010)*, pages 79–84. IOS Press, 2010.
- 5 Ronald Fagin, Joseph Y. Halpern, Yoram Moses, and Moshe Vardi. *Reasoning About Knowledge*. MIT Press, 1995.
- 6 J. Y. Halpern. A modification of the Halpern-Pearl definition of causality. In *Proc. 24th International Joint Conference on Artificial Intelligence (IJCAI 2015)*, pages 3022–3033, 2015.
- 7 J. Y. Halpern. *Actual Causality*. MIT Press, Cambridge, MA, 2016.
- 8 J. Y. Halpern and J. Pearl. Causes and explanations: A structural-model approach. Part I: Causes. *British Journal for Philosophy of Science*, 56(4):843–887, 2005.
- 9 H. L. A. Hart. The ascription of responsibility and rights. In Gilbert Ryle and Antony Flew, editors, *Proceedings of the Aristotelian Society*, pages 171–194. Blackwell, 1951.
- 10 J.L. Pollock. *Cognitive Carpentry*. The MIT Press, 1995.
- 11 H. Prakken. An abstract framework for argumentation with structured arguments. *Argument & Computation*, 1(2):93–124, 2010.
- 12 Ken Satoh. Private communication, 2018.
- 13 Ken Satoh and Satoshi Tojo. Disjunction of causes and disjunctive cause: a solution to the paradox of *conditio sine qua non* using minimal abduction. In Tom M. van Engers, editor, *Legal Knowledge and Information Systems - JURIX 2006: The Nineteenth Annual Conference on Legal Knowledge and Information Systems, Paris, France, 7-9 December 2006*, volume 152 of *Frontiers in Artificial Intelligence and Applications*, pages 163–168. IOS Press, 2006.
- 14 D. Walton. Evaluating expert opinion evidence. In *Argument Evaluation and Evidence*, volume 23 of *Law, Governance and Technology Series*, pages 117–144. Springer, 2016.
- 15 D. Walton and T. F. Gordon. Formalizing informal logic. *Informal Logic*, 35(4):508–538, 2015.

Participants

- Tobias Ahlbrecht
TU Clausthal, DE
- Natasha Alechina
University of Nottingham, GB
- Kevin D. Ashley
University of Pittsburgh, US
- Matteo Baldoni
University of Turin, IT
- Christoph Benz Müller
FU Berlin, DE
- Célia da Costa Pereira
Laboratoire I3S –
Sophia Antipolis, FR
- Mehdi Dastani
Utrecht University, NL
- Tiago de Lima
CNRS – Lens, FR
- Davide Dell’Anna
Utrecht University, NL
- Jürgen Dix
TU Clausthal, DE
- Ali Farjami
University of Luxembourg, LU
- Dov M. Gabbay
King’s College London, GB
- Aditya K. Ghose
University of Wollongong, AU
- Matthias Grabmair
Carnegie Mellon University –
Pittsburgh, US
- Joris Hulstijn
Tilburg University, NL
- Wojtek Jamroga
Polish Academy of Sciences –
Warsaw, PL
- Özgür Kafali
University of Kent –
Canterbury, GB
- Sabrina Kirrane
Wirtschaftsuniversität Wien, AT
- Brian Logan
University of Nottingham, GB
- Emiliano Lorini
University of Toulouse, FR
- Martin Neumann
Jacobs University Bremen, DE
- Pablo Noriega
IIIA – CSIC – Barcelona, ES
- Julian Padget
University of Bath, GB
- Adrian Paschke
FU Berlin, DE
- Nicolas Payette
LABSS – ISTC – CNR –
Rome, IT
- Ken Satoh
National Institute of Informatics –
Tokyo, JP
- Matthias Scheutz
Tufts University – Medford, US
- Viviane Torres da Silva
IBM Research –
Rio de Janeiro, BR
- Leon van der Torre
University of Luxembourg, LU
- Harko Verhagen
Stockholm University, SE
- Douglas Walton
University of Windsor, CA
- Michael Winikoff
University of Otago, NZ
- Vahid Yazdanpanah
University of Twente, NL

