Modelling Dynamic Normative Understanding in Agent Societies

Christopher K. Frantz[1], Martin K. Purvis[1], Bastin Tony Roy Savarimuthu[1], and Mariusz Nowostawski[2]

[1] Department of Information Science, University of Otago, New Zealand
{christopher.frantz,martin.purvis,tony.savarimuthu}@otago.ac.nz
[2] Faculty of Computer Science and Media Technology,
Gjøvik University College, Norway
mariusz.nowostawski@hig.no

Abstract. Agent-based Modelling offers strong prospects in the context of institutional modelling, which, from historical perspective, centres around the question of how far institutional instruments might have affected social and economic outcomes. To provide a richer representation of the institution formation process in the context of social simulation, we propose a norm generalisation process that uses an extended version of Crawford and Ostrom's institutional grammar and incorporates aspects from the area of social psychology. We believe that this approach offers a good compromise between generalisability and modelling detail. We briefly showcase this approach in the context of a scenario from the area of institutional economics to highlight its explanatory power.

Keywords: Norms, Institutions, Institutional Grammar, Norm Generalisation, Norm Synthesis, Dynamic Deontics, Maghribi Traders Coalition, Social Simulation, Agent-Based Modelling.

1 Introduction

Institutional modelling has received increasing attention in the area of multi-agent systems, and multi-agent-based simulation, such as in [9,1,20]. One central driver is the continued interest in explaining socio-economic development based on the institutional environment that either fostered or restrained economic development, which is a key theme of the area of New Institutional Economics [21]. Agent-based modelling is particularly useful in this context, since it can model human interaction on multiple levels of social organisation (micro, meso, macro).

In this connection our contribution concerns the development of a generalisable and accessible approach for the representation of institutions, here understood in their various forms, ranging from conventions, norms, to rules. In this work we augment an institutional representation structure, an extended version [6] of Crawford and Ostrom's institutional grammar [3], with a means of generalising norms from observed action experiences. By thus integrating 'structure' and 'process', we provide an integrated representation of social concepts

H.K. Dam et al. (Eds.): PRIMA 2014, LNAI 8861, pp. 294–310, 2014.

beyond the current precondition-deontic combination approach (see e.g. [14]). We illustrate the application of this mechanism using a specific scenario from the area of institutional economics.

In Section 2 we lay out the motivation in more detail. Section 3 provides a brief introduction of the institutional grammar, followed by the description of the norms generalisation process (Section 4). In Section 5 we apply the proposed mechanism to a simulation of our guiding scenario, concluding with a discussion and contextualisation of the contribution in Section 6.

2 Scenario Background and Motivation

To illustrate our present work, we employ a long-distance trading scenario metaphor inspired by the Maghribi Traders Coalition [11]. Under those arrangements, trade organisation was largely informal – traders delegated the transport and sale of goods to fellow traders in remote locations under the promise to reciprocate that service, an aspect that allowed them to expand the geographic range of their operations. Traders thus relied on mutual compliance; individuals that were suspected of embezzling profits, i.e. cheating, faced exclusion from trade.

In this society traders could at the same time adopt two roles: 1) *sender*, and 2) *operator*. Senders sent goods to other operators who then facilitated the actual sale and returned the realized profits to the sender.

We entertain a comparatively broad understanding of institutions [15,11], interpreting institutions as 'manifestations of social behaviour' that extend from conventions, via (informal) social norms, to (formal) rules. For this reason we seek to operationalise a general representation for institutions, such as that found in Crawford and Ostrom's Grammar of Institutions [3], which has been refined [6] to step beyond a descriptive perspective and support the modelling of emerging institutions. For our purposes the effectiveness of the 'grammar' lies in its human-readable interpretation, consideration of social structures (e.g. actors), as well as its cross-disciplinary applicability (see [19,18,9]). In the context of social simulation it can thus serve as an expressive interface between the experimenter and the observed artificial society. The present work's contribution is to augment this structural representation with a systematic process that describes how individuals can develop normative understanding based on generalised experience and observations.

3 Nested ADICO (nADICO)

The concept of Nested ADICO (nADICO) [6] builds on the essential purpose of the original institutional grammar [3] to represent conventions (or shared strategies[1]), norms and rules in the form of *institutional statements*. It consists

[1] A differentiation of 'shared strategies' beyond the notion introduced by Crawford and Ostrom [3] is discussed by Ghorbani et al. [8].

of five components (with the acronym ADICO), and is briefly explained in the following:[2]

- *A*ttributes (**A**) – describe the characteristics of individuals or groups of individuals that are subject to an institution;
- *D*eontics (**D**) – explicate whether the institutional statement is of prescriptive or permissive nature, originally based on deontic primitives (e.g. may, must, must not);
- A*I*m (**I**) – describes an action or outcome associated with the institutional statement;
- *C*onditions (**C**) – describe the circumstances under which a statement applies, which can include spatial, temporal and procedural aspects; and
- *O*r else (**O**) – describes consequences of non-compliance to a statement described by the above four components – 'Or else' itself can be a nested institutional statement.

This grammar allows for the expression of statements of varying nature and strength, representing different institution types, while allowing a straightforward transformation from natural language.

A convention, for example, can be adequately expressed using the components AIC, e.g.: **Traders (A) trade fair (I) when being observed (C).** Adding the *D*eontics component to the statement extends it to a norm statement: **Traders (A) *must* (D) trade fair (I) when being observed (C).** Finally, adding a consequence (*O*r else), constitutes a norm or a rule [6]:

Traders (A_0) must (D_0) trade fair (I_0) when being observed (C_0),
or else observers (A_1) must (D_1) report this (I_1) in any case (C_1).

Institutional statements can be *nested vertically* (as shown above), in which a consequential statement backs a statement it monitors (above: `ADIC(ADIC)`). Ideally this enables the modelling of comprehensive chains of responsibility. Beyond this, institutional statements can be *horizontally nested*, i.e. combined by logical operators that describe their co-occurrence (e.g. `ADIC` ***and*** `ADIC`) or alternative occurrence (inclusive or: `ADIC` ***or*** `ADIC`; exclusive or: `ADIC` ***xor*** `ADIC`). The formalised syntax is described in [6].

4 From Experiences to Institutions

Although conventions and norms surround us, we are often barely conscious of them and how they arise. Generally, norms are understood to be implicitly adopted on the basis of experiential [16] and social [2] learning in the contexts of existing institutions.

To follow this intuition concerning the subconscious development of such normative understanding, we employ a data-driven approach that utilizes the data structures we have described to facilitate agents' understanding of the normative environment involving a minimal amount of explicit reasoning. Although

[2] This elaboration is based on the extended grammar described in [6].

we present this model with respect to a specific problem, we believe that the proposed approach is generalisable and equips the modeller with a mechanism for norm representation that permits accessible inspection of simulations. Accordingly, we adopt a representation that has the descriptive power to capture instance-level actions as well as higher-level institutions (Section 4.1). Using this representation, we present a systematic process that describes the transition and derivation of institutional statements from individual observations (Section 4.2).

4.1 Action Representation

To put actions in a context for the purpose of instantiation, we can use the syntax of the grammar's AIC component that augments an action definition with the subject (*Attributes*) and context/conditions (*Conditions*). Utilising the term `act` to signify an individual action instance, an *action statement* is thus `act(attributes, aim, conditions)`, where `aim` represents the action definition.

Refining the *Attributes* component, we assume individuals to carry observable properties (attributes) that are equivalent to the social markers individuals display in real life, such as name, ethnic background, gender etc. We represent attributes as two sets, with the first set representing individual characteristics and the second highlighting group characteristics [12]. An example for the representation of attributes could thus be `attributes({id}, {role, ethnicity})`.

Furthermore, we need to specify a structured action specification in order to establish unambiguous symbolic references to an action and its properties. We define actions using a signifying term `a` and an associated set of properties `p`, the set of which depends on the nature of the action. Substituting the *aim* component of the institutional grammar, we can thus say `aim(a, p)`. Taking an example from our scenario, the central properties of the action 'send goods to trader' are the *object* that is dispatched ('goods'), as well as the destined *target* ('trader'). This *action definition* can thus be represented as `aim(send, {object, target})`, and instantiated as `aim(send, {goods, trader})`.

In addition we tailor the *Conditions* component to capture the context of action execution by allowing the specification of a potentially related preceding action (e.g. as a reaction to a previous action) as the first element, such as `conditions(act, *)`, along with potential further conditions.

Table 1 provides an overview of the refined component specifications.[3]

4.2 Generalisation

Individuals generally and unintentionally engage in processes of 'implicit social cognition' [10], one of which is the social generalisation process of 'stereotyping'. This process can lead to uncertainty reduction and efficiency enhancement, which is compatible with the purposes of institutions [15,22]. Stereotyping offers

[3] Note that as a matter of conciseness examples substitute the complete attributes specification of agents (i.e. including their social markers) with their name (e.g. Trader1).

Table 1. Component Specifications

Component	Structure	Example/Instance
Attributes	`attributes(i, s)`, with `i`/`s` being sets of individual/social attributes	`attributes({id}, {role})`
Action Definition	`aim(a, p)`, with `a` being a natural language action descriptor, and `p` being a set of action properties	`aim(send, {object, target})`
Conditions	`conditions(act, c)`, with `act` being a preceding action, and `c` being a set of further conditions	`act(Trader2, aim(trade, {goods}), conditions(act(Trader1, aim(send, {goods, Trader2}), *)))`
Action Statement	`act(attributes, aim, conditions)`	`act(Trader1, aim(send, {goods, Trader2}), *)`

individuals the ability to develop predictors to anticipate another's behaviour and to call upon previously executed successful reactions.

We model such processes based on the collected action information by applying a set of steps outlined in Figure 1 and described in the following.

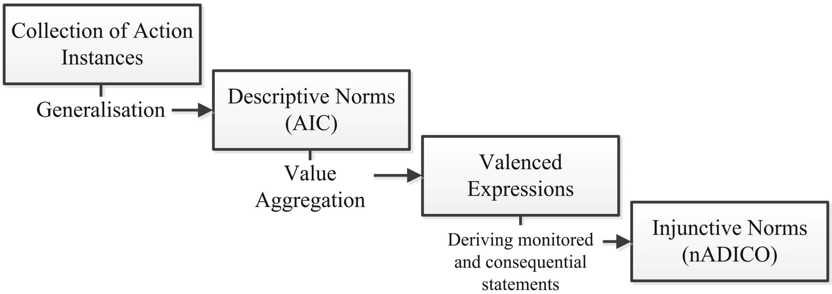

Fig. 1. Generalisation Process

Modelling subconscious generalisation processes shifts the perspective from the observation of individual behaviour instances to social behavioural regularities, closing the gap to what we perceive as institutions. We thus perform an aggregation of individual action statements to form generalised AIC statements, which we consider equivalent to observed conventions, or, in this case, *descriptive norms*. Operationally this is achieved by grouping the observed action statements based on their individual components, while keeping references to the action expressions constituting that respective AIC statement.

To explore this generalisation process, let us use a running example, a trade action instance. `Trader2` trades goods in the role of an `Operator`, after having been sent goods by another fellow trader `Trader1` (in the role of a `Sender`):

```
act(attributes({Trader2}, {Operator, Trader, Maghribi}),
    aim(trade, {goods}),
    conditions(act(attributes({Trader1}, {Sender, Trader, Maghribi}),
                   aim(send, {goods, attributes({Trader2}, {Operator, Trader, Maghribi})}),
                   conditions(*))))
```

Given the focal interest in actions, the totality of which express behavioural regularities, individual action statements are grouped based on the (decontextualised) action descriptor, i.e. the first element of an action statement's aim component. Referring to the running example this would be `trade`.

As a next step in the generalisation process, we consider the actor. Individuals base their generalisations on the social markers. In our example, the social marker with greatest relevance/salience in the context of trading is the role in which the individual operates (`Sender`/`Operator`). Ambiguous markers that describe supersets of the situationally relevant markers (here: `Trader` and `Maghribi`) are likewise maintained to serve for further generalisation (e.g. contrasting non-Maghribian traders from Maghribian ones, should such observation occur[4]) or to resolve conflicting statements.[5]

Finally, `attributes` components of actions held within the conditions component are likewise generalised to social markers, i.e. removing individual markers.

The generalisation process thus incurs the following steps:

1. Group all action statements (`act`) by action descriptor (`aim` component).
2. Group based on social markers by removing individual markers (`attributes` component).
3. Substitute all *preceding* action statements' attributes components (in `conditions` component) with social markers (i.e. remove individual markers as done in Step 2).

Assuming multiple statements showing the trading activity following the receipt of goods, we can express this as the generalised observation, or descriptive norm (`aic`), 'operators trade goods after having been sent goods by senders':

```
aic(attributes({Operator}),
    aim(trade, {goods}),
    conditions(aic(attributes({Sender}),
                   aim(send, {goods, attributes({Operator})}),
                   conditions(*))))
```

In order to develop more complex institutional statements beyond conventions or objectified descriptive norms, we need to assume that agents receive and associate feedback with individual action instances as part of their experiential learning process. Those then serve as input for the value aggregation process. The conceptualisation and implementation of feedback is domain/application-dependent and exemplified in Section 5.

4.3 Value Aggregation

The central purpose of the value aggregation process is to build up information used for an agent's overall understanding of a generalised convention. This is not

[4] Based on common marker subsets individuals could infer hierarchical conceptual relationships.

[5] For the following examples we will ignore the ambiguous markers.

to be confused with its attitude towards a convention, but instead represents the result of a cyclic internalisation and socialisation process which is based on experience and part of the agent's development of normative understanding. This aggregation operates on action instances grouped by the AIC statement. Table 2 shows simplified action instance representations for the previous generalisation, along with hypothetical feedback values.

Table 2. Exemplified action instances with valences

Simplified Action Example	Feedback
`act(trader1, trade, ...)`	30
`act(trader2, trade, ...)`	10
`act(trader3, trade, ...)`	−20
`act(trader4, trade, ...)`	20

For the aggregation we consider various *aggregation strategies*, possible approaches being the *summation of individual action feedback* to determine the *overall experience*; the *mean of feedback* represents a *rational conservative feedback expectation*; the *most extreme value* represents an *optimistic/pessimistic feedback expectation*.[6] The summation approach reflects the 'total experience', while the other measures discount feedback for a single action instance. Using the mean (in the example extract: 10) represents the rationally expected feedback. The aggregation based on the highest/lowest experience value, i.e. the individual's most extreme positive or negative experience (here: 30), puts emphasis on an individual's *most desired/feared* experience.

Ultimately, the aggregated value will be associated with the generalised AIC statement as a precursor for the development of nADICO statements. But, we first operationalise nADICO's *Deontics* component as a central mechanism to represent perceived duties, before deriving complete institutional statements.

4.4 Refining the Deontics Component

In contrast to the conventional characterisation of deontic primitives in discrete terms of prohibitions, permissions, and obligations, we apply a continuous notion of deontics, previously introduced as Dynamic Deontics [7], the essential intuitions of which are visualised in Figure 2.

In contrast to the discrete deontics understanding, a continuous representation of deontics can reflect the dynamic shifts between different deontics over time. Deontic terms, such as *must* and *may* are labels used for different deontic compartments along a deontic range, as shown in Figure 2. The tripartite nature of the deontic primitives demarks the midpoint and extremes of the deontic range, with intermediate compartments possibly labelled with terms[7] representing the

[6] One could introduce further aggregation mechanisms that include stronger weighting of recent or extreme values, or alternatively consider the variance of experiences.

[7] The choice of intermediate deontic terms for this example is not systematically grounded but follows the intuition of increasing prescriptiveness reaching from *may* to *must* and vice versa for proscriptions.

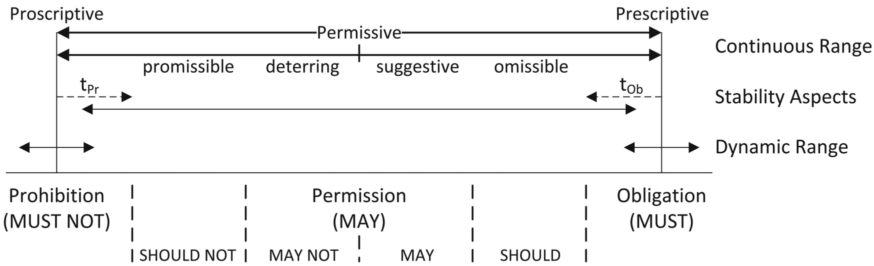

Fig. 2. Dynamic Deontic Range

gradual deontic notions of obligation and prohibition, such as the 'omissible' (obligations that can be foregone) or 'promissible' (prohibitions that can be ignored). The range itself is dynamic and determined by the individual's experience, a possible mapping being the direct association of the most positive and the most negative experiences with the respective ends of the deontic range. In the context of the current work, we operationalise this as the mean value across two memory instances holding sliding windows of memorized past highest and lowest aggregated values (see Subsection 4.3) for generalised action statements (see Subsection 4.2). Using the dynamically adjusting deontic compartments, the normative understanding of the individual, expressed as aggregated values, can be translated into semantically meaningful deontic labels.[8]

4.5 Deriving nADICO Expressions

To derive higher-level nADICO expressions from AIC statements, we revisit the developed action sequences that not only reflect an individual's actions but also multi-party actions. In addition the actors can be generalised based on their social markers, such as roles (e.g. sender of goods, recipient, etc.). Consequently agents cannot only derive behavioural conventions related to themselves, but, in principle, for any individual they observe, and further, predict individuals' behaviours based on existing social markers. This aspect is a precursor for applying cognitive empathy [4], such as the ability to perform perspective taking.

However, this mechanism requires the transformation of action sequences by *separating sequences into monitored statements and consequential statements* (see Section 3). The action sequences represent action/reaction pairs that suggest a 'because of', or 'on the grounds of' relationship. Using our example we would then arrive at the interpretation: "The operator trades goods *because* he has been sent goods by the sender.", which represents the descriptive norm perspective. However, to represent an injunctive perspective that highlights an individual's perception of its duties, we require the transformation into '*Or else*' relationships for cases in which the sequence's actors change (for example: '*Sender* sending goods', followed by '*Operator* trading goods.'). To reflect the injunctive nature

[8] A more comprehensive overview over concept and motivation is provided in [7].

of the '*Or else*', we attach representations of perceived duty (deontics) and **invert** the derived consequential statement's deontic ('Senders *have to* send goods, or else Operators *will NOT* trade goods.'). Again, note that the deontic terms associated with the deontic range (*may*, *should*, *must*) may not precisely reflect this understanding, but they capture the intuition of such expression.

To proceed along these lines, it is necessary to distinguish separate action sequences originating in one's social environment. To do this, an agent identifies the first preceding statement whose attributes (i.e. generalised social markers) differ from the last statement's attributes. Using this approach, an agent can discriminate between actions and possible reactions. The aggregated value derived in the previous step can then be associated with the identified monitored statement's *Deontics* component.

To establish the deontic term's matching counterpart, the individual's existing deontic range facilitates the inversion of the aggregated value, the result of which is assigned to the consequential statement. For example, assuming a deontic range with midpoint value of 0, a value of 5 on a situational range – perhaps mapping to *should* – is inverted to its opposite scale value and deontic term (*should not*). Applying our running example, the corresponding nADICO statement reads:

```
adico(attributes({Sender}),
      deontics(5),
      aim(send, {goods, attributes({Operator})}),
      conditions(*),
      orElse(adic(attributes({Operator}),
                  deontics(-5),
                  aim(trade, {goods}),
                  conditions(*))))
```

Let us summarize the algorithmic steps for this approach:

1. Store last generalised action's *Attributes*.
2. Starting from the last generalised action, iterate through preceding generalised action statements (*previousAction* in *Conditions* component) until either finding a statement whose *Attributes* differ or no further preceding statement exists.
   - If statement with different *Attributes* is found, consider subsequence processed prior to current iteration as *consequential statement*; the tested statement and the remaining subsequence are assigned as *monitored statement*.
   - If no differing *Attributes* are found in preceding generalised action statements[a], treat first processed action statement (i.e. last action of action sequence) as *consequential statement*; the subsequence of preceding action statements is treated as *monitored statement*.
3. Assign aggregated value (see Subsection 4.3) to *Deontics* component of *monitored statement*.
4. If a *consequential statement* exists, invert the aggregated value (from Step 3) on the deontic range and assign it to the consequential statement's *Deontics* component.

[a] In this case all elementary actions of a sequence have been performed by agents of identical social markers. The last elementary action is then treated as previous actions' consequence.

Note that this derivation approach does not establish a consequential statement if no previous action has been observed, generating an injunctive norm without specified consequences.

At this stage, the derived nADICO statements provide the experimenter with a comprehensive insight into individuals' experience-based normative understanding. Moreover, the derived statements can be further generalised based on individual components, such as an overall normative understanding of actions (aims) by aggregating nADICO statements for specific actor perspectives (e.g. Operator, Sender), as alluded to in Subsection 4.2, or for a particular action.

We will explore this mechanism using the simulation scenario described at the outset of this article.

5 Simulating Normative Understanding within Maghribi Trader Society

To demonstrate how trading can develop normative understanding, we describe a model in which agents do so based on environmental feedback.

5.1 Model

Traders establish a maximum number (*maxRelationships*) of mutual trade relationships to other traders based on random requests, to which they then send goods. The receiver trades those goods at a profit that is determined by a random factor between *minProfit* and *maxProfit*, with the market being represented by a random number generator. If initialised as cheater, the trader cheats with a probability $p_{cheating}$ and otherwise returns the profit to the original sender. Sending agents memorize the returned profits (as action feedback) in a memory holding a parametrised number of last entries, which they can query for specific individuals or across all their partners in order to gauge the correctness of the returned profit. In cases of presumed cheating, traders can fire the suspect and memorize it as a cheater. The interactions represent actions of the structure introduced in Subsection 4.1, with preceding actions stored in the *Conditions* component, successively building up action sequences that represent the comprehensive transactions between individual agents.

Naturally, the randomly generated profit (which we assume to be positive on average) introduces fuzziness into the decision-making of profit recipients (the original goods senders). To accommodate the fluctuation of returns, the sender's evaluation mechanism compares the operator's performance with its previous record. In the default strategy, operators are only fired if they produce negative profits and their mean previous returns are likewise negative. In all other cases agents are considered non-cheaters. To reflect the ongoing trade relationship, rewards are represented as the trade value of previous interactions with the rewarding party (i.e. the profits the other trader had generated for the service provider over time). If the operator's cheating was not detected, the embezzled fraction is added as part of his reward. In the case of firing, the inverted trade value of that partner is memorized along with the action sequence.

Agents can differentiate between private and public action sequences, with private action sequences containing additional actions, such as cheating (from the perspective of a cheating operator)[9] or suspected cheating (from the perspective of the original sender of the goods), which the agents memorize but do not share.

Based on their experience agents derive normative statements as outlined in the previous sections. Traders utilise the derived statements to guide their decisions whether to continue sending goods and to return profits, by aggregating them based on the given actions (e.g. sending goods, returning profit) across one's overall experience. Given our characterisation of norms as continua, traders have an individualised tolerance towards aggregated negative feedback (*defectionThreshold*), which is randomly determined at the time of initialisation and lies between zero and *maxDefectionThreshold*. Such tolerances ultimately limit market interactions and thus affect the overall economic performance.

Algorithm 1 outlines the agents' execution cycle; Algorithm 2 specifies agents' reactions to incoming requests.

Algorithm 1. Agent Execution Cycle

```
if < maxRelationships relationships to other traders then
    Pick random trader this agent does not have relationship with
        and send relationship request;
    if request is accepted then
        Add accepting trader to set of relationships;
if agent has relationships to other traders then
    Pick random fellow trader from set of relationships;
    if normative understanding of action is above defectionThreshold then
        Send goods to selected trader and await return of profit;
        if profit < 0 and mean value of memorized past transactions from
        trader is < 0 then
            Extend received action statement with action cheat;
            Memorize trader as cheater;
            Fire trader;
        else
            Reward trader;
        end
        Memorize action statement in association with profit;
Derive nADICO statements from memory;
```

5.2 Evaluation

We initialise the simulation with the parameters outlined in Table 3 and use the number of performed transactions per round as a target variable to indicate overall economic performance.

[9] It would hardly be useful if an agent were to report his cheating to the goods' owner. Instead he would merely indicate that he traded the goods, but, depending on feedback, internalise if his cheating (in combination with trading) was successful.

Algorithm 2. Agent Reactions

Initialisation: Initialise agent as cheater with probability $p_{cheater}$;
Case 1: Receipt of relationship request
if *requester is not memorized as cheater and* $<$ *maxRelationships relationships* **then**
    Accept request and add requester to relationships;
else
    Reject request;
end
Case 2: Receipt of trade request
Perform market transaction;
if *initialised as cheater* **then**
    Determine whether to cheat (based on $p_{cheating}$);
    if *cheating* **then**
        Determine random fraction f of profit to embezzle ($0 \leq$ f ≤ 1);
        Create private copy of action statement and add action `cheat`;
        Memorize extended action sequence;
if *normative understanding of action is above defectionThreshold* **then**
    Return profit as part of publicly visible action statement (not indicating whether cheated or not);
Case 3: Receipt of firing notification
Remove sender from own relationships;
Mark sender as cheater;

Table 3. Simulation Parameters

Parameter	Value	Parameter	Value
Number of agents	100	$p_{cheater}$ (Fraction of cheaters)	0.2
maxRelationships	8	$p_{cheating}$	0.6
minProfit (Factor of goods' value)	0.8	*maxDefectionThreshold*	-100
maxProfit (Factor of goods' value)	1.3	Number of memory entries	100

Given the fuzziness in which traders determine cheating, the simulation parameters were refined after repeated operational runs to minimize the observation of false positives in the absence of cheating and to offer stable transaction levels, oscillating between 180 and 200 transactions per round (i.e. two transactions per trader per round – one as sender, one as operator). The generalised nADICO statements, along with deontic terms derived from the mapping of values can then be observed for individual agents as shown in Figure 3.

The statements show an overview of the agent's normative understanding, but also highlight potentially emerging conflicts, such as Statements 2 and 4, which are generalised based on the different reactions the agent experiences, in one case[10] indicating that it *must* return profit after he has been sent goods, traded and cheated, or otherwise not receive rewards. Given that he received a reward

[10] Read Statement 2 as (actions emphasized): 'Operators must *return profits* if they have *traded* goods they have been *sent* (by senders), and *cheated* while trading, or else senders will not *reward* them.'

for his trade (i.e. his cheating was not discovered), cheating appears to be a desirable action. Statement 4 shows generalisation based on the rare case that his cheating *was* discovered and sanctioned with dismissal (see the *O* component). Using the representation with dynamic deontics, different statements can be clearly prioritised: the higher deontic value in Statement 2 indicates that trading and cheating (see action sequence in conditions component) is more attractive, compared to mere trading shown in Statement 3 (Note the lower deontic value). Statements can likewise be integrated based on their common action sequence (e.g. by addition of deontic values). For example, Statements 2 and 4 (that have a common action sequence, but different consequences ('REWARD' vs. 'FIRE') and deontic values) can help explain why agents favour cheating (Statement 2) despite the (low) risk of being fired (Statement 4).

```
L0: A=A(*, [SENDER]), D=75.0 (MUST), I=I(SEND_GOODS, *), C=C(*),
    O=(L1: A=A(*, [OPERATOR]), D=-146.10054 (inv) (MUST NOT), I=I(RETURN_PROFIT, *), C=C({PREVIOUS_ACTION=
       A=A(*, [OPERATOR]), I=I(TRADE, *), C=C(*)}), O=(null))                                          1

L0: A=A(*, [OPERATOR]), D=79.25663 (MUST), I=I(RETURN_PROFIT, *), C=C({PREVIOUS_ACTION=
 (L0: A=A(*, [OPERATOR]), I=I(TRADE, *), C=C({PREVIOUS_ACTION=
  L0: A=A(*, [SENDER]), I=I(SEND_GOODS, *), C=C(*), O=(null)}), O=(null)) AND                           2
 (L0: A=A(*, [OPERATOR]), I=I(CHEAT, *), C=C(*), O=(null))}),
    O=(L1: A=A(*, [SENDER]), D=-154.39249 (inv) (MUST NOT), I=I(REWARD, *), C=C(*), O=(null))

L0: A=A(*, [OPERATOR]), D=26.537308 (SHOULD), I=I(RETURN_PROFIT, *), C=C({PREVIOUS_ACTION=
 L0: A=A(*, [OPERATOR]), I=I(TRADE, *), C=C({PREVIOUS_ACTION=                                           3
  L0: A=A(*, [SENDER]), I=I(SEND_GOODS, *), C=C(*), O=(null)}), O=(null)}),
    O=(L1: A=A(*, [SENDER]), D=-51.694866 (inv) (SHOULD NOT), I=I(REWARD, *), C=C(*), O=(null))

L0: A=A(*, [OPERATOR]), D=-1.6361282 (INDIFFERENT), I=I(RETURN_PROFIT, *), C=C({PREVIOUS_ACTION=
 (L0: A=A(*, [OPERATOR]), I=I(TRADE, *), C=C({PREVIOUS_ACTION=
  L0: A=A(*, [SENDER]), I=I(SEND_GOODS, *), C=C(*), O=(null)}), O=(null)) AND                           4
 (L0: A=A(*, [OPERATOR]), I=I(CHEAT, *), C=C(*), O=(null))}),
    O=(L1: A=A(*, [SENDER]), D=0.8398984 (inv) (INDIFFERENT), I=I(FIRE, *), C=C(*), O=(null))
```

Fig. 3. Situational nADICO Statements for individual agent

In order to gain a society-wide perspective on the normative landscape, we can analyse the distribution of individual normative understandings across the deontic range. We visualise this using a Kiviat-inspired chart that shows distributions across adjacent ordinally scaled values, such as deontic terms. Figure 4 shows monitored statements aggregated by leading attributes (i.e. acting role) and aim components at around 1,900 trading rounds.[11] At that stage traders are split whether or not it is worthwhile continuing to send goods (36 say *may* and 46 say *must not*) based on continuous cheating. Acting as operators, all traders maintain the understanding that processing received goods and returning them is worthwhile, indicating that they are generally rewarded. Only a subset of operators (the black series) perceives cheating as rewarding.

[11] Each statement is represented as a separate series. The axes' lengths are scaled relative to the deontic term with greatest support (here: 46).

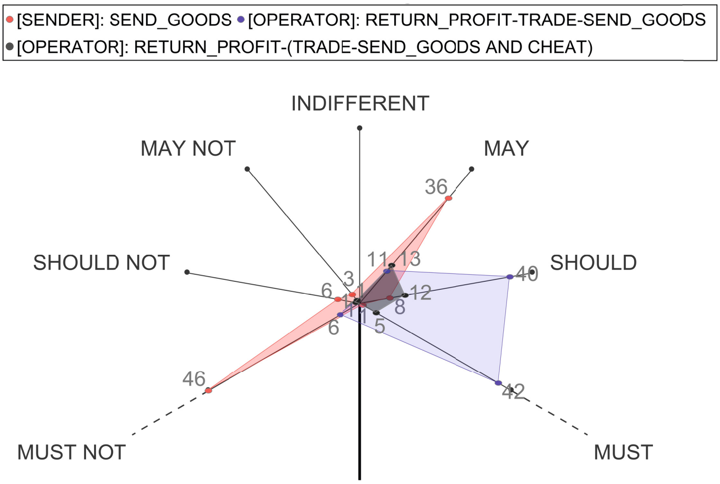

Fig. 4. Distribution of Normative Understanding after 1,900 rounds

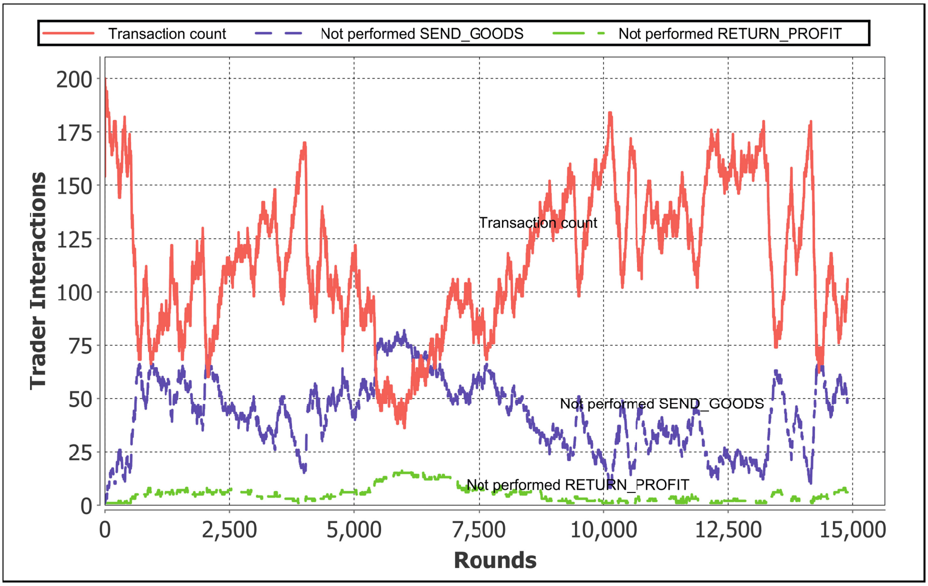

Fig. 5. Number of trade interactions; defections from actions over time

To provide a dynamic perspective of this changing normative landscape over time, we provide a link to a video showing the evolution of the successive deontic charts (including a second chart focusing on cheaters) [5].

From Figure 5 we can observe an initially high commitment of traders to engage in trade interactions that starts to fluctuate once a sufficient degree of cheating is experienced. This is based on the understanding that sending goods is likely to be followed by cheating (Series 'SEND_GOODS'), and to a lesser extent, that returning goods honestly is sanctioned by firing (Series 'RETURN_PROFIT'). Trade is restored once those traders have only acted as operators for some time, erasing (by gradually forgetting) the negative experience associated with their operation as senders of goods. The parameter set explored here thus shows a borderline case between a well-functioning trader society and economic collapse caused by cheating. However, when higher numbers of traders also reject the returning of goods (e.g. at around 6,000 rounds), trading comes close to a collapse. The cheating probability is a central parameter in this simulation set, with lower values maintaining a functioning trade system, and values > 0.6 accelerating the oscillation even further. Increasing the number of traders, in contrast, reduces the amplitude of trade variations and thus increases economic stability.

6 Discussion and Outlook

We have provided a candidate operationalisation for norm emergence based on an expressive institutional grammar. Its operation has been demonstrated by means of a multi-agent trade scenario. The nADICO structure offers a detailed and unified representation of institutions, encompassing differentiated action structures, but, perhaps more importantly, fostering a multi-perspective representation of actions (here in the form of different roles). We believe that this grammar represents a suitable combination offering a) a human-readable representation that allows direct interaction with the experimenter, and b) highly expressive syntax that captures action combinations and sequences, action subjects, context, nesting of statements and various institution types. The grammar can be directly used, in conjunction with the process steps laid out in this paper, for normative modelling with a minimal set of prior specifications to be derived from the modelled application scenario (social markers, action specification, feedback).

This work fits well into the research field of normative modelling [1,20], with recent emphasis on norm synthesis that captures aspects of norm generalisation. However, in contrast to other approaches from the area of norm synthesis, such as [13,14], our approach does not require the specification of an explicit ontology to drive norm generalisation, but may well infer hierarchical conceptual relationships based on common social marker subsets (see Section 4.2) while still offering a richer syntax for norm representation. An important aspect of norm synthesis is the treatment of norm conflicts. Recent work on robust self-governing systems by Riveret et al. [17] relies on explicit concensus mechanisms to resolve norm conflicts. Our approach does not require such mechanisms. Instead, the numerical representation of 'oughtness' using the concept of Dynamic Deontics allows for a mathematical integration of conflicting perceived duties – recall the conflicting motivations to embezzle profits, with the carrot of being rewarded and the stick of being fired (Statements 2 and 4 in Figure 3).

Given our focus on the norm derivation process, the experimental model itself has not been explored to its full extent in this text. Nevertheless, the evaluation highlights the essential features of the generalisation process and showcases the interpretation of emerging norms. We also constrained the sensing capabilities for this scenario to experiential learning. However, the model itself is by no means limited to this type of learning, but could likewise incorporate social learning as well as direct communication. In fact, the action representation derived from the nADICO syntax (Section 4.1) can well serve as a message container for inter-agent communication, including (but not limited to) norm representation.

We believe that this modelling of norms is truthful to their actual nature. The approach assumes a consequentialist perspective in which we do not presume pre-imposed norms (though those may certainly exist and could be predefined), but drives normative understanding purely based on behaviouristic principles and without explicit sharing of norms. This supports their interpretation as shared implicit behavioural regularities, while maintaining an unambiguous representation that allows a flexible analysis based on arbitrary characteristics (e.g. for separate roles, specific actions, and different social groups/structures).

References

1. Andrighetto, G., Villatoro, D., Conte, R.: Norm internalization in artificial societies. AI Commun. 23(4), 325–339 (2010)
2. Bandura, A., Ross, D., Ross, S.A.: Transmission of aggression through imitation of aggressive models. J. of Abnormal and Social Psychology 63(3), 575–582 (1961)
3. Crawford, S.E., Ostrom, E.: A Grammar of Institutions. In: Understanding Institutional Diversity, ch. 5, pp. 137–174. Princeton University Press, Princeton (2005)
4. Decety, J., Grèzes, J.: The power of simulation: imagining one's own and other's behavior. Brain Research 1079(1), 4–14 (2006)
5. Frantz, C.: Video of Evolving Normative Understanding based on Dynamic Deontics, `https://unitube.otago.ac.nz/view?m=zhQu6cTOdOj` (accessed: September 2014)
6. Frantz, C., Purvis, M.K., Nowostawski, M., Savarimuthu, B.T.R.: nADICO: A Nested Grammar of Institutions. In: Boella, G., Elkind, E., Savarimuthu, B.T.R., Dignum, F., Purvis, M.K. (eds.) PRIMA 2013. LNCS, vol. 8291, pp. 429–436. Springer, Heidelberg (2013)
7. Frantz, C., Purvis, M.K., Nowostawski, M., Savarimuthu, B.T.R.: Modelling Institutions Using Dynamic Deontics. In: Balke, T., Dignum, F., van Riemsdijk, M.B., Chopra, A.K. (eds.) COIN 2013. LNCS, vol. 8386, pp. 211–233. Springer, Heidelberg (2014)
8. Ghorbani, A., Aldewereld, H., Dignum, V., Noriega, P.: Shared Strategies in Artificial Agent Societies. In: Aldewereld, H., Sichman, J.S. (eds.) COIN 2012. LNCS, vol. 7756, pp. 71–86. Springer, Heidelberg (2013)
9. Ghorbani, A., Bots, P., Dignum, V., Dijkema, G.: MAIA: a Framework for Developing Agent-Based Social Simulations. Journal of Artificial Societies and Social Simulation 16(2) (2013)

10. Greenwald, A.G., Banaji, M.R.: Implicit social cognition: attitudes, self-esteem, and stereotypes. Psychological Review 102, 4–27 (1995)
11. Greif, A.: Institutions and the Path to the Modern Economy. Cambridge University Press, New York (2006)
12. Haslam, S.A., Ellemers, N., Reicher, S.D., Reynolds, K.J., Schmitt, M.T.: The social identity perspective today: An overview of its defining ideas. In: Postmes, T., Branscombe, N.R. (eds.) Rediscovering Social Identity, pp. 341–356. Psychology Press (2010)
13. Morales, J., Lopez-Sanchez, M., Rodriguez-Aguilar, J.A., Wooldridge, M., Vasconcelos, W.: Automated synthesis of normative systems. In: AAMAS 2013 (2013)
14. Morales, J., Lopez-Sanchez, M., Rodriguez-Aguilar, J.A., Wooldridge, M., Vasconcelos, W.: Minimality and simplicity in the automated synthesis of normative systems. In: AAMAS (2014)
15. North, D.C.: Institutions, Institutional Change, and Economic Performance. Cambridge University Press, Cambridge (1990)
16. Parsons, T.: The Social System. Routledge, New York (1951)
17. Riveret, R., Artikis, A., Busquets, D., Pitt, J.: Self-governance by transfiguration: From learning to prescriptions. In: Cariani, F., Grossi, D., Meheus, J., Parent, X. (eds.) DEON 2014. LNCS, vol. 8554, pp. 177–191. Springer, Heidelberg (2014)
18. Siddiki, S., Weible, C.M., Basurto, X., Calanni, J.: Dissecting Policy Designs: An Application of the Institutional Grammar Tool. The Policy Studies Journal 39, 79–103 (2011)
19. Smajgl, A., Izquierdo, L.R., Huigen, M.: Modeling endogenous rule changes in an institutional context: The ADICO Sequence. Advances in Complex Systems 2(11), 199–215 (2008)
20. Villatoro, D., Andrighetto, G., Sabater-Mir, J., Conte, R.: Dynamic sanctioning for robust and cost-efficient norm compliance. In: IJCAI 2011, vol. 1, pp. 414–419. AAAI Press (2011)
21. Williamson, O.E.: Markets and Hierarchies, Analysis and Antitrust Implications: A Study in the Economics of Internal Organization. Free Press, New York (1975)
22. Williamson, O.E.: Transaction Cost Economics: How it works; Where it is headed. De Economist 146(1), 23–58 (1998)