

Self-Organising Common-Pool Resource Allocation and Canons of Distributive Justice

Jeremy Pitt, Julia Schaumeier, Dídac Busquets and Sam Macbeth

Department of Electrical & Electronic Engineering

Imperial College London, Exhibition Road, London, SW7 2BT, UK

Email: {j.pitt, j.schaumeier09, didac.busquets, samuel.macbeth06}@imperial.ac.uk

Abstract—Two principles of enduring institutions for self-organising resource allocation are congruence of the allocation method to the resources available, and participation of those affected by the allocation (the appropriators) in selecting that method. However, the principles do not say anything explicitly about the *fairness* of the allocation method, or the outcomes. In this paper, we complement these principles with canons of distributive justice represented as *legitimate claims*, which are implemented as voting functions that determine the order in which resource requests are satisfied. The appropriators vote on the weight attached to the scoring functions, and so self-organise the allocation method. Experiments with a variation of the Linear Public Good game show that this pluralistic self-organising approach produces a better balance of utility and fairness (for agents that comply with the rules of the game) than monistic or fixed approaches.

Keywords—self-organising systems, common-pool resource allocation, electronic institutions, distributive justice.

I. INTRODUCTION

Open computing systems, from sensor networks to Smart-Grids, face the same challenge: a set of autonomous, heterogeneous actors, or agents, needing to collectivise and distribute resources without a centralised decision-making authority. This challenge is further complicated by an *economy of scarcity*, where there are fewer resources available than are required in total.

There are many scenarios where a group of (human) actors need to pool their resources to achieve individual goals. A simple game-theoretic analysis predicts that such actors will inevitably deplete the resource; for example, the incentive to use the resource before someone else leads to pumping races in water management scenarios. Ostrom [1], however, maintained that such depletion was far from inevitable, and cited many cases where the resource was sustained, even over many generations, without centralisation, privatisation or other intervention by external authorities. In all these cases, the actors had formed an institution, defining a set of rules which conventionally regulated and constrained provision to and appropriation from the resource, i.e. self-governing the commons.

Ostrom also observed that in some cases the institution was formed but the resource still was not sustained. She then identified eight principles for enduring institutions, which

were necessary and sufficient conditions for the institution to maintain the resource. A meta-review has corroborated these principles with only minor clarifications [2].

Two of these principles for enduring institutions for self-organising resource allocation are congruence of the allocation method to the resources available and the environment, and participation of those affected by the allocation in selecting that method. The first principle ensures that the resource allocation method is appropriate to the state of the environment, including the amount of resources available (cf. process scheduling in operating systems – in ‘normal’ loads, a priority queue can be used; with ‘heavy’ loads, a random schedule is no worse than other methods, and saves on expending resources on determining the priority). The second principle ensures some form of participation, whereby those affected by the rules have a say in the selection of the rules, either by consultation, vote, etc.

However, the principles do not say anything explicitly about the *fairness* of the allocation method, or the outcomes. In this paper, we complement these two principles with a resource allocation method based on canons of *distributive justice* [3]. We review the background to principles for enduring institutions and canons of distributive justice in Section II. In Section III, we introduce a variant of the Linear Public Good game [4] as an abstract representation of the resource allocation scenarios found in ad hoc networks, sensor networks, SmartGrids, etc. The canons of distributive justice are then represented, in Section IV, as *legitimate claims*, which are implemented as voting functions that determine the order in which resource requests are satisfied.

A formal model of a self-organising institution, in which appropriators (agents) vote on the weight attached to the scoring functions, and so self-organise the allocation mechanism is presented in Section V. Experimental results reported in Section VI show that this pluralistic self-organising approach produces a better balance of utility and fairness (for agents that comply with the rules of the game, according to the applied fairness metric) than monistic or fixed approaches. As a result we unify principles of enduring self-organising institutions with canons of distributive justice to provide a basis for designing mechanisms to address the resource allocation problem in open systems.

II. INSTITUTIONS AND ‘FAIR’ RESOURCE ALLOCATION

A. Self-Governing Institutions

In her work on self-governing institutions for common-pool resource management, Ostrom defined such an institution as “a set of working rules that are used to determine who is eligible to make decisions in some arena, what actions are allowed or constrained, ... [and] contain prescriptions that forbid, permit or require some action or outcome” [1, p. 51]. The “working rules” specified procedures for operational-, collective- and constitutional-choice, and were respectively concerned with provision, appropriation and monitoring; determining the operational-choice rules, rule enforcement and dispute resolution; and eligibility for determining the collective-choice rules.

These rules were role-based, mutually agreed, mutable and nested within each other in *decision arenas* or *action situations*. Distinguishing between nested situations requires a formal characterisation of institutionalised power [5], whereby an agent appointed to (or occupying) a role in a particular action situation is empowered to bring about a fact of conventional (or institutional) significance by performing a designated action in that specific context. A role might be fixed in one action situation, but could be changed by rules of another action situation within which it was nested.

Ostrom also observed that on some occasions the resource was not depleted, and on others it was. Eight principles were identified as necessary and sufficient conditions for a common-pool resource managed by a self-governing institution to *endure* [1, p. 90], as listed in Table I.

Table I: Ostrom’s Principles for Enduring Institutions.

1	Clearly defined boundaries.
2	Congruence between appropriation and provision rules and the state of the prevailing local environment.
3	Collective-choice arrangements: those affected by the operational-choice rules participate in selection and modification of those rules.
4	Monitoring by accountable agencies.
5	Graduated sanctions.
6	Access to fast, cheap conflict-resolution mechanisms.
7	Existence of and control over their own institutions is not challenged by external authorities.
8	Systems of systems.

In this work, we are primarily concerned with Ostrom’s Principles 2 and 3, and addressing the issue that, given a resource allocation decided by rules that are congruent with the environment and determined by collective-choice arrangements, *is it fair?*

There are many quantitative metrics for calculating ‘fairness’ objectively, but for *self*-governing institutions we start from Rescher’s analysis of *distributive justice* [3].

B. Distributive Justice

Rescher’s analysis of distributive justice concludes that the Principle of Utility, taken as a fairness metric expressed

as “the greater good of the greater number”, is but one of many prevailing considerations which need to be taken into account when determining a ‘fair’ allocation of resources. Rescher then argued that an adequate theory of distributive justice requires coordination of the concepts of justice, construed in terms of fairness and equity, and of utility, in the sense of general welfare.

Rescher observed that distributive justice had been held, by various sources, to consist of treating people wholly or primarily according to one of seven *canons* (established principles expressed in English). These canons are summarised in Table II, as the canons of equality, need, ability, effort, productivity, social utility and supply & demand.

Table II: Rescher’s Canons of Distributive Justice.

1	Treatment as equals.
2	Treatment according to their needs.
3	Treatment according to their actual productive contribution.
4	Treatment according to their efforts and sacrifices.
5	Treatment according to a valuation of their socially-useful services.
6	Treatment according to supply and demand.
7	Treatment according to their ability, merit or achievements.

Rescher’s analysis showed that each canon, taken in isolation, was inadequate as the sole dispensary of distributive justice. Instead, his position was that distributive justice was found in the *canon of claims*, which consists of treating people according to their legitimate claims, both positive and negative. This, Rescher claimed, placed the emphasis for distributive justice onto the questions of: what the legitimate claims are, how they are accommodated in case of plurality, and how they are reconciled in case of conflict.

Our position is that a self-governing institution should take into account not only the state of the ‘environment’, but also Rescher’s idea of *legitimate claims* to those resources [3, p. 81ff]. In the next section, we define an endogenous resource provision and appropriation scenario, and in the subsequent section consider the representation of legitimate claims in that context, and how to deal with plurality and reconciliation.

III. THE LINEAR PUBLIC GOOD GAME

The problem of individual resource contribution to a system for collective action has been analysed using the linear public good (LPG) game [4]. However, the LPG game makes several assumptions, so to study resource allocation in open systems, we utilise a variation, *LPG'* [6].

In an *LPG'* game, n agents form a *cluster*. The game itself is played in consecutive rounds, $t_0, t_1, \dots, t_\infty$ (we omit identifying a round by a subscript t if it is clear from context). In each round t , each (player) agent i :

- Determines the resources it has available, $g_i \in [0, 1]$.
- Determines its need for resources, $q_i \in [0, 1]$ ($q_i > g_i$).
- Makes a demand for resources, $d_i \in [0, 1]$.

- Makes a provision of resources, $p_i \in [0, 1]$ ($p_i \leq g_i$).
- Receives an allocation of resources, $r_i \in [0, 1]$.
- Makes an appropriation of resources, $r'_i \in [0, 1]$.

The LPG' game assumes an *economy of scarcity*, by insisting that each agent's need for resources is greater than those it is capable of generating for itself, i.e. $q_i > g_i$ for each agent i . Thus agents are necessarily dependent on others, there is an incentive not to comply with the rules, and the resource allocation method has to cope with the deficiency.

The total resources accrued by an agent at the end of a round is given by R_i :

$$R_i = r'_i + (g_i - p_i)$$

i.e. R_i is the sum of the resources that are *appropriated* (rather than allocated) from the common-pool and the available resources that are withheld from the pool. The utility of agent i is then given by:

$$U_i = \begin{cases} a(q_i) + b(R_i - q_i) & \text{if } R_i \geq q_i \\ a(R_i) - c(q_i - R_i) & \text{otherwise} \end{cases}$$

where a , b and c are coefficients in \mathbb{R} measuring, respectively, the relative utilities of getting resources that are needed, getting resources that are not needed, and not getting resources that are needed. Note that appropriated resources do not accrue from one round to the next.

Independent of its utility, each agent i in cluster C makes a *subjective* assessment of its *satisfaction* $\sigma_{i,C}$, represented as a value in $[0, 1]$, based on its allocation in relation to its demands. Each agent increases its satisfaction in the next round if it is allocated at least the same as its demand in the current round, and decreases it otherwise:

$$\sigma_{i,C}(t+1) = \begin{cases} \sigma_{i,C}(t) + \alpha \cdot (1 - \sigma_{i,C}(t)) & \text{if } r_i \geq d_i \\ \sigma_{i,C}(t) - \beta \cdot \sigma_{i,C}(t) & \text{otherwise} \end{cases}$$

where α and β are coefficients in $[0, 1]$ which determine the rate of reinforcement of, respectively, satisfaction and dissatisfaction. We also define a threshold or cut-off value τ and an interval value m , such that if for m consecutive rounds, an agent i evaluates $\sigma_i(C, t) < \tau$ as true, then it will leave the cluster C . It will not re-join a cluster it has previously left.

Note that the five coefficients (a, b, c, α, β) and the satisfaction threshold values (τ, m) need not be the same for every agent, but for the experiments in Section VI we will use global values.

There are many different ways of determining the set of allocations r_i based on the set of demands d_i [7]. For example, a dictatorial solution might satisfy the demand of the dictator first, then allocate what is left equally to each remaining member. A competitive solution could be found based on auctions [8]. A more collaborative solution could be based on giving an exactly equal share to each of the n agents. More elaborate algorithms can refine rationing by re-allocating an excess (i.e. if $r_i > d_i$) to an agent with

the greatest need. Each of these approaches has its strengths and weaknesses, so in this work we will investigate a self-organising allocation scheme based on *legitimate claims*.

IV. LEGITIMATE CLAIMS IN LPG'

The 'paradoxical' nature of the LPG' game is similar to the LPG. That is, the incentive is to withhold provision, mis-represent need, and appropriate more than is allocated, thereby maximising individual utility and satisfaction. If all agents behave like this, then each agent is reduced to relying on its own available resources, satisfaction decreases, and eventually the cluster disintegrates as players leave the game through dissatisfaction.

Therefore, it is necessary to design a mechanism to incentivise provision, encourage accurate representation of needs, and discourage excess appropriation. We do so by defining an allocation mechanism using Rescher's canons of distributive justice, based on legitimate claims [3] within a self-governing institution, also taking into account the quantity of resources to be allocated, the method by which resources are to be allocated, the outcome of the method as a mapping from the members of the institution to an actual allocation, and the monitoring of their behaviour.

In the context of a self-governing institution for endogenous resource provision and appropriation according to the LPG' game, we note that there individual facts made public, i.e. the demand, d_i and the allocation r_i . We assume that the the provision p_i and appropriation r'_i are monitored, and 'everyman' satisfaction σ_i (i.e. what the satisfaction would be for an 'ordinary' agent if it had a generic α and β , irrespective of its actual, internal satisfaction). The basic role of *prosumer* determines which agent is and is not a member of the institution, and there is one distinguished role, the *head*, who is responsible for the allocation.

Using this information, the legitimate claims, which determine the relative merit of the member's claims, are defined by the following functions. T denotes the total number of rounds of the LPG' game played in a particular cluster C , and $T_{\{i \in C\}}$ denotes the number of rounds that agent i has played in cluster C .

f_1 : *The canon of equality* – there are three ways of representing the claim: first (f_{1a}) rank the agents in increasing order of their average allocations:

$$\frac{\sum_{t=0}^T r_i(t)}{T_{\{i \in C\}}}$$

secondly (f_{1b}) rank the agents in increasing order of their 'everyman' satisfaction σ_i ; and thirdly (f_{1c}), rank the agents in increasing order of the number of rounds in which they received an allocation:

$$\frac{\sum_{t=0}^T (r_i(t) > 0)}{T_{\{i \in C\}}}$$

f_2 : *The canon of needs* – assuming for similar agents that demand averages out over time, then the agent most in need is the one that has hitherto made the least demand, therefore rank the agents in increasing order of their average demands:

$$\frac{\sum_{t=0}^T d_i(t)}{T_{\{i \in C\}}}$$

This also encourages accurate reporting of demands.

f_3 : *The canon of productivity* – rank the agents in decreasing order of their average provision.

$$\frac{\sum_{t=0}^T p_i(t)}{T_{\{i \in C\}}}$$

Note that for this claim, there are several alternatives, for example using the total provision ($\sum_{t=0}^T p_i(t)$) rather than the average provision, or net provision (the difference between provision and allocation):

$$\frac{\sum_{t=0}^T p_i(t) - \sum_{t=0}^T r_i(t)}{T_{\{i \in C\}}}$$

f_4 : *The canon of effort* – rank the agents in decreasing order of the number of rounds spent in the *prosumer* role of the cluster, participating in the *LPG'* game, i.e. $T_{\{i \in C\}}$.

f_5 : *The canon of social utility* – rank the agents in decreasing order of the number of rounds spent in a distinguished role of the cluster, e.g. the *head*.

f_6 : *The canon of supply & demand* – let us suppose there are multiple clusters, each playing its own instance of the *LPG'* game. The players most ‘in demand’ are those agents who are most compliant, i.e. those agents who follow the ‘norms’ of the game. These norms are firstly, do not withhold what is available ($p_i = g_i$); secondly, only demand what is needed ($d_i = q_i$); and thirdly, only appropriate what is allocated ($r'_i = r_i$). Therefore, for this canon, rank the agents according to this measure of compliance:

$$\text{card}(\{t \in T_{\{i \in C\}} \mid (p_i(t) = g_i(t)) \wedge (d_i(t) = q_i(t)) \wedge (r'_i(t) = r_i(t))\})$$

This assumes that we can monitor every agent’s internal state, monitoring and reporting is perfect, and the cost of monitoring is zero. The effect of relaxing these assumptions, in the context of retributive justice, is investigated in [6].

f_7 : *The canon of merits and achievements* – is not appropriate in this context and is not represented.

This deals with the question of representing legitimate claims. To accommodate multiple claims, each canon f_* is treated as a *voter* in a Borda count protocol. Under Borda count voting, each vote ranks the list of candidates in order of preference. Borda points are assigned to each candidate in the list: for example, with n candidates, rank k scores $n - k + 1$ Borda points. Borda points from each vote are summed to give a total Borda score for each candidate.

In our case, each voting function f_* rank orders all the agents in a candidate list according to the relative grounds for their claims, and the Borda score for each agent is computed from the accumulation of Borda points associated with each vote. Normally, in a Borda count protocol, the candidate with the highest Borda score wins, but we may have multiple ‘winners’, so we form a Borda point queue in descending order of Borda score, and allocate resources to the front of the queue until there are no more to allocate.

To reconcile conflicts between multiple claims, a *weight* $w_* \in [0, 1]$ is attached to each function $f_* \in F$. The Borda score B of agent i under a set of functions F is given by:

$$B(i, F) = \sum_{*=1}^{|F|} w_* \cdot \text{bpts}(f_*(i))$$

where $f_*(i)$ computes the rank order assigned to agent i by each f_* , $\text{bpts}()$ computes the Borda points for that rank, and w_* is the weight attached to the corresponding function.

The final procedure is specified in Algorithm 1. The function `Borda_count` computes the Borda score B for each agent i and returns the monotonic list *Borda_ptq*.

Algorithm 1 Resource Allocation with Legitimate Claims.

```

1:  $A =$  set of  $n$  agents,
2:  $F =$  set of  $m$  voting functions  $f_*$  each with weight  $w_*$ 
3: for each agent  $i \in A$  do
4:   demand  $d_i$ , provision  $p_i$ 
5: end for
6:  $P \leftarrow \sum_{i=1}^n p_i$ 
7:  $\text{rank\_orders} \leftarrow []$ 
8: for each function  $f_* \in F$  do
9:    $\text{rank\_orders} \leftarrow \text{rank\_orders} \cup f_*(A)$ 
10: end for
11:  $\text{Borda\_ptq} \leftarrow \text{Borda\_count}(\text{rank\_orders}, F)$ 
12: repeat
13:    $i \leftarrow \text{head}(\text{Borda\_ptq})$ 
14:    $\text{Borda\_ptq} \leftarrow \text{tail}(\text{Borda\_ptq})$ 
15:   if  $P \geq d_i$  then
16:      $r_i \leftarrow d_i, P \leftarrow P - d_i$ 
17:   else
18:      $r_i \leftarrow P, P \leftarrow 0$ 
19:   end if
20: until  $P == 0$ 

```

Determining the weight associated with each claim is an outcome of *self-organisation*: we propose that the agents themselves determine the weights using the collective-choice rules of a self-organising electronic institution.

V. SELF-ORGANISING ELECTRONIC INSTITUTIONS

In this section, we will apply the same sorts of ‘moves’ as in [9], to define a self-organising electronic institution for resource provision and appropriation in the *LPG'* game.

First, we define a formal model of the institutional structure. Then we give a logical axiomatisation of the operational-choice rules in the framework of dynamic norm-governed multi-agent systems [10]. Finally, we consider the ‘agent-internal’ reasoning involved in expressing a preference for the weighting on the canons of distributive justice, which are required to self-organise the relative importance of the individual claims.

A. Formal Model

In an institutional setting, n agents form a cluster to play the LPG' game in an action situation, nested within an overall institution. There may be multiple action situations within the institution. We will say that an agent occupies the role of *prosumer* in a particular action situation if it is playing the LPG' game in that situation.

Let \mathcal{IC}_t be a multi-agent system at time t defined by:

$$\mathcal{IC}_t = \langle \mathcal{A}, \mathcal{C}, \mathcal{L}, G, F \rangle_t$$

where (omitting the subscript t if clear from context):

- \mathcal{A} is the set of all agents;
- \mathcal{C} is the set of clusters (action situations);
- \mathcal{L} is a norm-governed system specification;
- G is the LPG' game;
- F is the set of voting functions representing legitimate claims.

In the framework of [10], a number of degrees of freedom (DoF) are identified, so \mathcal{L} defines a specification space, where each specification instance is defined by a different set of values assigned to the DoF. For example, one degree of freedom is the method by which the resources are allocated (e.g., at random, by ration, or using legitimate claims), and another might be, if using legitimate claims, how many claims and which ones.

Each cluster (action situation) $C_t \in \mathcal{C}_t$ is given by:

$$C_t = \langle \mathcal{M}, l, \epsilon \rangle_t$$

where (again omitting the subscript t if clear from context):

- \mathcal{M} is the set of prosumer agents, such that $\mathcal{M} \subseteq \mathcal{A}$
- l is a specification instance of \mathcal{L} ; and
- ϵ is the cluster’s local environment, a pair $\langle Bf, If \rangle$

Regarding the environment ϵ , Bf represents the set of ‘brute’ facts whose values are determined by the *physical* state, including the sum of common-pool resources P as a result of provision by the agents. If represents the set of ‘institutional’ facts, whose values are determined by the *conventional* state, i.e. are asserted by the exercise of institutionalised power.

The institution’s operational-choice rule, as specified by Algorithm 1, maps a set of agents (encapsulating each of their demands, provisions, etc.) and a set of voting functions to an allocation:

$$ocr : \mathcal{M} \times F \rightarrow (\mathcal{M} \mapsto [0, 1])$$

The agents themselves determine the weights associated with each of the functions in F , using the collective-choice rule:

$$ccr : V(\cdot)_{a \in \mathcal{M}} \times F \rightarrow F$$

so that ccr takes a set of expressed preferences (votes) from the agents in the action situation and a set of existing weights on the voting functions F , and computes a new set of weights according to the votes.

The rest of this section is concerned with the logical axiomatisation (specification) of these two institutional rules.

B. Logical Axiomatisation of Operational-Choice

The rules of \mathcal{L} can be defined by an executable specification using the Event Calculus (EC) [11]. The EC is a logical formalism for representing and reasoning about actions or events and their effects based on a many-sorted first-order predicate calculus. An *action description* in EC includes axioms that define a narrative (the occurrence of actions), using the happensAt predicate; the effects of actions, using initiates and terminates predicates; and the values of the fluents, using initially and holdsAt predicates. A fluent is then a proposition whose values can change over time.

In the sequel, we specify an EC action description using axioms of the form:

$$\begin{array}{l} \textit{Action} \text{ initiates } F = V \text{ at } T \leftarrow \\ \textit{Conditions} \end{array}$$

which are read as stating: the occurrence of action *Action* at time T initiates a period of time for which the value of fluent F is V , if the *Conditions* are satisfied.

The physical and institutional facts in an action situation’s environment ϵ are represented as EC fluents. Analysing the LPG' game, there are, for each agent i :

- Two *internal* values, available resources g_i and needed q_i , whose values are not represented in the environment.
- One *declared* value, demand d_i , which is an institutional fact whose value is initiated by each empowered agent i , i.e. a *prosumer* in the action situation.
- One *computed* value, allocation r_i , which is an institutional fact whose value is determined by the resource allocation method and initiated by an empowered agent occupying the role of the *head* of the action situation.
- Two *physical* values, provision p_i and appropriation r'_i , whose values are determined by monitoring or observation of the corresponding actions.

We also need fluents to represent the roles of each agent, the (institutionalised) powers, permissions and obligations of each agent, the resources in the common pool P , the weights associated with the voting functions in each action situation, and the outcome of applying the Borda count protocol.

The actions that the agents can perform in the role of *prosumer* are *demand*, *provide*, *appropriate* and *vote*. The agent occupying the role of *head* is empowered to perform

the *declare* and *allocate* actions. In the parameter list of actions (and fluents), A and H are (instantiated to) agents, R to a value in $[0, 1]$ representing resource amount, C to a cluster (action situation), P to the common pool, and L to the *Borda_ptq* computed by Algorithm 1 at timepoint T .

A *demand* action initiates d_i to the demanded value if performed by an empowered agent.

$$\begin{aligned} & \text{demand}(A, R, C) \text{ initiates } d(A, C) = R \text{ at } T \leftarrow \\ & \quad \mathbf{pow}(A, \text{demand}(A, R, C)) = \text{true} \text{ holdsAt } T \\ & \mathbf{pow}(A, \text{demand}(A, R, C)) = \text{true} \text{ holdsAt } T \leftarrow \\ & \quad \text{role}(A, C) = \text{prosumer} \text{ holdsAt } T \end{aligned}$$

A *provide* action contributes to the pool P . Note that this is a physical action so empowerment is not an issue. It is required to represent both institutional value of the pool *ifpool* for computing the allocation and the physical value of the pool *bfpool* for performing the appropriation. For simulation, it is assumed that agents do not provide more than they have available ($p_i \leq g_i$).

$$\begin{aligned} & \text{provide}(A, R, C) \text{ initiates } \text{ifpool}(C) = P' \text{ at } T \leftarrow \\ & \quad \text{ifpool}(C) = P \text{ holdsAt } T \wedge \\ & \quad P' = P + R \end{aligned}$$

and similarly for *bfpool*(C). This implements lines 3–6 of Algorithm 1.

The *head* agent then applies the voting functions for legitimate claims, and declares the result as the value of the fluent *Borda_ptq*:

$$\begin{aligned} & \text{declare}(H, L, C) \text{ initiates } \text{Borda_ptq}(C) = L \text{ at } T \leftarrow \\ & \quad \mathbf{pow}(H, \text{declare}(H, L, C)) = \text{true} \text{ holdsAt } T \\ & \mathbf{pow}(H, \text{declare}(H, L, C)) = \text{true} \text{ holdsAt } T \leftarrow \\ & \quad \text{role}(H, C) = \text{head} \text{ holdsAt } T \end{aligned}$$

This implements lines 7–11 of Algorithm 1.

An allocation of resources to an agent is initiated by the empowered agent (the agent appointed to the role of *head* of the cluster) performing the designated action. This initiates new values for three fluents: the allocation r_i to the agent, the reduced pool *ifpool* of resources to allocate, and removal of agent from the front of *Borda_ptq*.

$$\begin{aligned} & \text{allocate}(H, A, R, C) \text{ initiates } r(A, C) = R \text{ at } T \leftarrow \\ & \quad \mathbf{pow}(H, \text{allocate}(H, A, R, C)) = \text{true} \text{ holdsAt } T \\ & \text{allocate}(H, A, R, C) \text{ initiates } \text{ifpool}(C) = P' \text{ at } T \leftarrow \\ & \quad \mathbf{pow}(H, \text{allocate}(H, A, R, C)) = \text{true} \text{ holdsAt } T \wedge \\ & \quad \text{ifpool}(C) = P \text{ holdsAt } T \wedge \\ & \quad P' = P - R \end{aligned}$$

$$\begin{aligned} & \text{allocate}(H, A, R, C) \text{ initiates } \text{Borda_ptq}(C) = Tl \text{ at } T \leftarrow \\ & \quad \mathbf{pow}(H, \text{allocate}(H, A, R, C)) = \text{true} \text{ holdsAt } T \wedge \\ & \quad \text{Borda_ptq}(C) = [A \mid Tl] \text{ holdsAt } T \end{aligned}$$

$$\begin{aligned} & \mathbf{pow}(H, \text{allocate}(H, A, R, C)) = \text{true} \text{ holdsAt } T \leftarrow \\ & \quad \text{role}(H, C) = \text{head} \text{ holdsAt } T \wedge \\ & \quad \text{Borda_ptq}(C) = [A \mid Tl] \text{ holdsAt } T \wedge \\ & \quad \text{ifpool}(C) = P \text{ holdsAt } T \wedge \\ & \quad d(A, C) = R \text{ holdsAt } T \wedge R \leq P \\ & \mathbf{pow}(H, \text{allocate}(H, A, R, C)) = \text{true} \text{ holdsAt } T \leftarrow \\ & \quad \text{role}(H, C) = \text{head} \text{ holdsAt } T \wedge \\ & \quad \text{Borda_ptq}(C) = [A \mid Tl] \text{ holdsAt } T \wedge \\ & \quad \text{ifpool}(C) = R \text{ holdsAt } T \wedge \\ & \quad d(A, C) = R' \text{ holdsAt } T \wedge R' \geq R \end{aligned}$$

This implements lines 12–20 of Algorithm 1.

Finally, there is the appropriation action:

$$\begin{aligned} & \text{appropriate}(A, R, C) \text{ initiates } \text{bfpool}(C) = P' \text{ at } T \leftarrow \\ & \quad \text{bfpool}(C) = P \text{ holdsAt } T \wedge \\ & \quad P \geq R \wedge P' = P - R \\ & \text{appropriate}(A, R, C) \text{ initiates } r'(A, C) = R \text{ at } T \leftarrow \\ & \quad \text{bfpool}(C) = P \text{ holdsAt } T \wedge \\ & \quad P \geq R \end{aligned}$$

Note that appropriation is a physical action but the agents can check that $r_i = r'_i$, i.e. that the appropriation was the same as the allocation.

C. Self-Organisation and Autonomy of Collective-Choice

At the end of each round of the *LPG'* game, the agents self-organise the weights on the voting functions. There are three motivations to be considered in this self-organisation: maximising self-interest, avoiding path dependency, and restoring a homeostatic equilibrium. Therefore we define three functions for computing new weights. The first function assumes that the agents vote for self-interest (i.e. for the functions that give them each the highest rank). The second function aims to resist entrenched self-interest and avoid path dependency based on the ‘distance’ between the voting function’s output and the actual allocation. Finally, the third function applies an autonomic mechanism via a form of ‘institutional homeostasis’ on the legitimate claims.

Each voting function f_* outputs a rank order of the agents, which is effectively a vector represented by \mathbf{f}_* . We denote by $\text{idx}_C(i, \mathbf{f}_*)$ the index position of each agent i participating in the cluster C in vector \mathbf{f}_* .

Assuming that each agent i votes for its preference for which function should have more weight in the next round according to its position in the rank order of this round, so each agent i ’s vote is a rank order of the functions, which is effectively a vector represented by \mathbf{i} . Note that:

$$\text{idx}_C(f_x, \mathbf{i}) < \text{idx}_C(f_y, \mathbf{i}) \rightarrow \text{idx}_C(i, \mathbf{f}_x) \leq \text{idx}_C(i, \mathbf{f}_y)$$

The Borda points for each function is given by the sum of the indices in \mathbf{i} of the functions which gave that agent the

same position, divided by the number of those functions. For example, suppose we had four functions which ranked agent i second, third, third and fourth. Then agent i 's vote would give the first function 4 Borda points, the second and third functions 2.5 Borda points each (i.e. a half share of $3 + 2$), and the fourth function 1 Borda point. We denote by $Borda(f, i)$ the Borda points assigned to function f by the vote of i .

For each function $f_* \in F$, its total Borda points in cluster C is given by:

$$Borda(f_*, C) = \sum_{i \in C} Borda(f_*, i)$$

and the updated weight w_* of each f_* is given by:

$$w_*(t) = w_*(t) + w_*(t) \cdot \frac{Borda(f_*, C) - AvgBorda}{TotalBorda}$$

A voting function which does better than average will get a higher weight in the next round, while those that do worse than average will get a lower weight. Note that the new weights are summed and normalised to 1.

To counter the self-interest that agents will continue to vote for (and increase the weight on) functions that work best for them, the weight on functions that are closer to the final rank order ($Borda_ptq(C)$) is reduced, while for those functions that were less 'accurate' the weight is increased. This is computed by taking the hamming distance (hd) between \mathbf{f}_* and $Borda_ptq(C)$ and updating the weights on the functions in the next round by:

$$w_*(t) = w_*(t) + w_*(t) \cdot \frac{hd(\mathbf{f}_*, Borda_ptq(C)) - AvgHD}{TotalHD}$$

Note again that all the new weights are summed and normalised to 1.

Finally, assuming that the 'equilibrium state' to be one that gives equal weights to each legitimate claim, then we want to restore that state. If all the agents in the cluster comply with the rules, i.e. $|C|$ is the same size as the set that comply according to canon f_6 , then each function weight is moderated as follows:

$$\begin{aligned} w_*(t+1) &= w_*(t) - (\gamma * (w_*(t) - \frac{1}{|F|})), \text{ if } w_*(t) > \frac{1}{|F|} \\ &= w_*(t) + (\gamma * (\frac{1}{|F|} - w_*(t))), \text{ if } w_*(t) \leq \frac{1}{|F|} \end{aligned}$$

Here γ is a coefficient controlling the autonomic mechanism.

VI. EXPERIMENTAL RESULTS AND EVALUATION

This section describes the implementation of a testbed and experimental results, evaluating the performance of a self-organising institution for playing the LPG' game according to the canons of distributive justice.

A. Testbed Implementation

The testbed was written using the Presage2 platform (see <http://www.presage2.info>). The control loop for the testbed is shown in Algorithm 2.

For each decision arena, one agent is assigned to the *head* role in each arena. A single iteration of the loop, playing one round of the LPG' game, consists of interleaving *updates* of internal agent states with *performance* of external actions, which change the environmental state.

Each agent has a fixed probability of cheating $pCheat$ each round and a cheating strategy *cheatOn*. The latter denotes which action they will cheat on, *provision*, *demand* or *appropriate*. A compliant agent i will provision what they generate ($p_i = g_i$), demand what they need ($d_i = q_i$), and appropriate what they are allocated ($r'_i = r_i$). The cheating behaviour for each of these actions is as follows:

- $p_i = g_i * rand(0, 1)$ - Reduces the usual provision (equal to the resources generated by the agent this round) by a random factor between 0 and 1.
- $d_i = q_i + rand(0, 1) * (1 - q_i)$ - Randomly increases the quantity demanded above what the agent needs.
- $r'_i = r_i + rand(0, 1) * (1 - r_i)$ - Increases the quantity appropriated above what the agent has been allocated.

At specific times (in the experiments we used every 20 cycles) the agents re-assign their roles in the cluster.

Algorithm 2 Testbed for LPG' game.

```

1: generate_agents(M, A)
2: generate_decision_arenas(N, A, C)
3: t ← 0
4: repeat
5:   for each decision arena C ∈ C do
6:     update cluster roles, agent status
7:     for each agent i ∈ C do
8:       update available resources gi, need ni
9:     end for
10:    compute Borda_ptq(C) using Algorithm 1
11:    for each agent i ∈ C do
12:      perform appropriation
13:      update utility Ui, satisfaction σi
14:      express preference on F
15:    end for
16:    compute new w*(f*, C) for each f* ∈ F
17:  end for
18:  t ← t + 1
19: until t = Tlim

```

To evaluate the fairness of the decision arena for cluster C for a given run, we use the Gini inequality index over $\left\{ \sum_{i=0}^T allocate_{ij} / \sum_{i=0}^T demand_{ij} : j \in 1 \dots |C| \right\}$. An index of 0 is perfect equality (i.e. fairness), and 100 is complete inequality.

B. Experimental Results

For our experiments the following parameter values were used unless stated otherwise: $a = 2$, $b = 1$, $c = 3$, $\alpha = 0.1$, $\beta = 0.1$, $\gamma = 0.1$, $\tau = 0.1$, $m = 3$, $\sigma_{i,C}(0) = 0.5$. In all experiments the agents only cheated on provision, however the results for cheating on demand and appropriation produced comparable results. 50 repeats were run for each experiment, with each repeat given a unique random seed.

In the first set of experiments, we compared the performance of each of the canons of legitimate claims individually (f_*) and all together (f_{all}), with f_{1a} , f_{1b} and f_{1c} being distinct claims. f_{all} has two variants, ‘fix’ which gives all distinct claims equal weight ($w_*(t) = 0.125$) and ‘SO’ which uses self-organised function weights. We used a population of 30 agents, each with a fixed probability p_{Cheat} of non-compliance with the rules of the LPG' game. In this experiment, there were 20 ‘compliant’ ($p_{Cheat} = 0.02$) and 10 ‘non-compliant’ ($p_{Cheat} = 0.25$) agents, for $T_{lim} = 500$ rounds of the LPG' game.

The results (Table III) show that, for this population:

- f_{1a} , f_{1b} , f_{1c} , f_4 and f_{all} (fix) produce an enduring cluster where all agents remain (r.), but benefit the non-compliant (NC) agents;
- f_5 produces a partially enduring cluster, heavily biased towards those that manage to occupy a role;
- f_2 , f_3 and f_6 do not produce an enduring cluster (remaining agents $r.(C) + r.(NC) = 0$);
- f_{all} (SO) produces an enduring cluster which benefits the compliant agents (C).

Table III: Comparison of Legitimate Claims

voting fn	utility C	utility NC	r. C	r. NC	fairness C
f_{1a}	71 ± 5	144 ± 7	20	10	0.73 ± 0.12
f_{1b}	74 ± 4	140 ± 6	20	10	0.63 ± 0.10
f_{1c}	71 ± 4	143 ± 6	20	10	0.69 ± 0.10
f_4	72 ± 7	142 ± 11	20	10	1.92 ± 0.29
f_5	61 ± 45	127 ± 88	8	4	15.61 ± 1.50
f_2	14 ± 14	25 ± 22	0	0	29.11 ± 4.12
f_3	37 ± 5	-25 ± 7	0	0	25.10 ± 4.73
f_6	50 ± 6	-33 ± 2	0	0	15.77 ± 2.14
f_{all} (fix)	89 ± 8	112 ± 16	20	10	1.02 ± 0.19
f_{all} (SO)	140 ± 28	8 ± 54	20	7	1.61 ± 0.67

Figure 1 shows how the weights on each canon develop in a game with the f_{all} (SO) allocation. We see that the compliant majority’s preferences cause higher weights on f_3 and f_6 , both of which penalise cheating agents (recall that agents cheat on provision so a lower average provision, as measured by f_3 , implies more cheating). Further experiments have shown that self-organisation is robust to different cheating strategies, i.e. the weights adjust to favour whichever canons discriminate against non-compliant agents.

In a second set of experiments, we compared an allocation strategy based on legitimate claims with self-organised

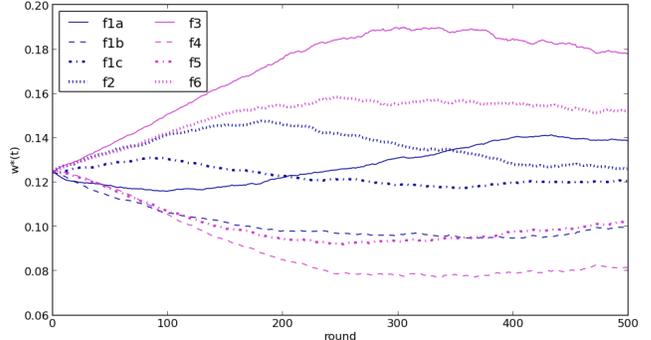


Figure 1: Individual function weights for f_{all} (SO)

weights against strategies based on fixed weights, random order and strict ration (equal allocation). We used three populations of agents: (1) a heterogenous population of 30 agents identical to the first experiment; (2) a homogenous population of 30 compliant agent ($p_{Cheat} = 0$) with a ‘low’ dissatisfaction reinforcement ($\beta = 0.1$); and (3) a homogenous population of 30 compliant agent ($p_{Cheat} = 0$) with a ‘high’ dissatisfaction reinforcement ($\beta = 0.4$). Other parameters were as in the first experiment.

The results (Table IV) show that, for these settings:

- an allocation strategy based on equal ration is ineffective in an economy of scarcity;
- a strategy based on random allocation benefits the non-compliant agents in a heterogenous population, and does not produce an enduring cluster at high dissatisfaction reinforcement rates in a homogenous one;
- when there are no non-compliant agents there is no gain from self-organisation and legitimate claims with fixed weights is sufficient for the cluster to endure;
- an allocation strategy based on legitimate claims with self-organised weights benefits the compliant agents (heterogenous) and produces a nearly fully enduring cluster (homogenous);
- the self-organising allocation strategy is fairest for the compliant agents according to our chosen fairness metric.

Figure 2 compares f_{all} (SO) and $random$ over a larger range of population compositions. With a random allocation the utility of non-compliant agents is always higher than the utility of the compliant players, however the self-organising legitimate claims allocation reverses this while the compliant agents are the majority.

In a third set of experiments, we tested the effect of a choice of clusters on a mixed distribution of agents. We used a population of 40 agents, with 20 compliant ($p_{Cheat} = 0.02$) and 20 non-compliant ($p_{Cheat} = 0.25$), for $T_{lim} = 1500$ rounds of the LPG' game. The agents are initially equally distributed over two clusters; one cluster (cl.1) using f_{all} (SO), the other cluster (cl.2) using $random$. If an agent’s

Table IV: Fairness of Allocation Methods

strategy	ut. C	ut. NC	r. C	r. NC	fairness C
ration het01	3 ± 2	11 ± 2	0	0	2.64 ± 0.55
// hom01	7 ± 1	n/a	0	n/a	2.60 ± 0.33
// hom04	1 ± 1	n/a	0	n/a	5.36 ± 0.70
random het01	72 ± 6	142 ± 10	20	10	1.83 ± 0.29
// hom01	124 ± 5	n/a	30	n/a	1.77 ± 0.27
// hom04	36 ± 9	n/a	0	n/a	6.25 ± 1.87
fixed het01	89 ± 7	114 ± 12	20	10	1.01 ± 0.31
// hom01	124 ± 5	n/a	30	n/a	0.76 ± 0.16
// hom04	123 ± 5	n/a	30	n/a	0.79 ± 0.13
SO het01	147 ± 24	-3 ± 43	20	7	1.59 ± 0.54
// hom01	124 ± 5	n/a	30	n/a	0.74 ± 0.13
// hom04	124 ± 4	n/a	29	n/a	0.83 ± 0.21

Table V: Self-Organisation into Clusters

β -value	ut. C	ut. NC	r. C	r. NC	fairness C
0.1 in cl.1	198 ± 55	90 ± 52	11	3	11.33 ± 5.25
// cl.2	35 ± 25	204 ± 40	9	17	12.74 ± 6.02
0.4 in cl.1	143 ± 71	52 ± 33	6	1	16.07 ± 5.62
// cl.2	3 ± 3	7 ± 2	0	0	14.87 ± 4.22

C. Evaluation with respect to Related Research

There are a large number of fairness theories in the social sciences (including Rescher [3] and Rawls [12]), and some of this work has been successfully transposed to multi-agent systems via Computational Social Choice for distributed resource allocation and negotiation [13]. A specific concern is to define metrics and design computational models that encourage (or compel) rational agents to determine an optimal or fair allocation of resources (i.e. mechanism design).

However, these approaches remain monistic, with only one fairness criteria being used at any given time. Moreover, as pointed out by De Jong [14], some of the designed fairness models cannot be actually realised, either because of their computational complexity, or due to the fact that one cannot force autonomous agents to behave fairly. This latter observation is very relevant to open systems, where the agents’ behaviour is not controlled by the system’s designer, which creates opportunities for non-compliant agents that may not be willing to show act according to the system rules. Even in De Jong’s work, which tries to avoid such shortcomings, the resulting model is still monistic.

Although the idea of combining different monistic criteria is not completely new, this has been limited to using one fairness criteria with a utilitarian one. As far as we are aware, there are no methods that combine different fairness criteria. Moreover, in the fairness-utilitarian combination, the weight of each of the two criteria is fixed, thus not allowing any form of self-organisation to change them.

A set of axioms that define some families of fairness measures is presented in [15], including many of the previously described metrics in other works (e.g. Gini, Jain, Atkinson indices, entropy, α -fair utility), and are also able to capture the principles and theories of fairness coming from social sciences. However, the work presents *how to define* fairness measures, but not *how to achieve* such fair allocations.

In future work, we will investigate a wide range of different objective fairness measures to examine whether Rescher’s legitimate claims yield a fair allocation irrespective of the fairness metric. However, these fairness metrics evaluate the system’s performance from a global perspective only, and do not take individual fairness perceptions into account. In human institutions, (perceived) *interactional injustice* (e.g. ‘unfair’, unequal treatment of individuals) affects motivation and impedes the productivity of these institutions. In order to see the effect on electronic institutions,

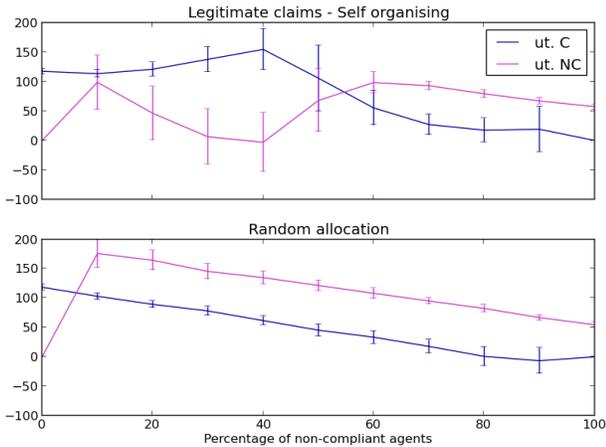


Figure 2: Avg. utility for C and NC in f_{all} (SO) and *random*

satisfaction falls below its recorded satisfaction the other cluster for m consecutive rounds it will change cluster.

Two runs were compared, one with ‘low’ dissatisfaction reinforcement ($\beta = 0.1$), and one with ‘high’ dissatisfaction reinforcement ($\beta = 0.4$).

The results (Table V) show that, for these settings:

- at low dissatisfaction reinforcement, we see a partial separation of agents, with the compliant agents preferring the cluster with legitimate claims, and the non-compliant agents in the cluster with random order;
- the cluster with legitimate claims benefits the compliant agents more than non-compliant;
- at high dissatisfaction reinforcement, cl.2 does not endure, and cl.1 partially endures and is virtually devoid of non-compliant agents.

To conclude, we argue that these results are an empirical (computational) validation of Rescher’s approach to distributive justice. Self-organisation combined with resource allocation according to pluralistic legitimate claims is no less fair (given the presented fairness metric) than alternatives, but much more beneficial to compliant agents, in an economy of scarcity where the violation of rules is possible.

we will experiment with agents that choose the weights on legitimate claims according to their individual fairness metric, offering a richer self-organisation than one based purely on self-interest.

We will further integrate these mechanisms for distributive justice with principles for *retributive justice*. These principles include agent monitoring and appropriate punishment measures for detected non-compliant agents, which will also serve to mitigate the effects of interactional injustice.

VII. SUMMARY AND CONCLUSIONS

In summary, this work builds on the foundations of self-organising electronic institutions established in [9], which showed that Ostrom’s design principles were necessary and sufficient conditions for enduring electronic institutions, and its stand-alone contributions are:

- it defines a new variant of the LPG game, *LPG'*, which relaxes some assumptions, changes the game payoffs and introduces a subjective assessment of satisfaction;
- it formalises another theory of fundamental socio-economic principles, namely Rescher’s canons of distributive justice represented by legitimate claims;
- it uses the framework of dynamic norm-governed systems to specify a formal model of self-organising institutions with multiple action situations involving the provision and appropriation of endogenous resources;
- it implements a new testbed which allows agents to self-organise the formalisation of canons of distributive justice, implemented as weighted voting functions; and
- it reports a new set of experimental results which show that such pluralistic electronic institutions lead to greater fairness than alternative approaches.

In conclusion, in terms of mechanism design, it could be argued that the preference of compliant agents for the cluster that computes resource allocation using self-organised legitimate claims demonstrates that this is precisely a mechanism which incentivises compliance and guarantees fairness. On the other hand, there are many experimental parameters and random factors in the testbed, and some substantive areas for further investigation (especially in the inter-dependence of distributive justice with procedural, interactional and retributive justice), so perhaps a weaker conclusion, that this is an ‘existence proof’ showing it is possible to construct a game with the desired properties, is warranted.

However, we showed in [9] that the ‘optimal’ short-term distribution of resources was less important than the sustainability of the resource itself, and that sustainability was a product of self-organisation. As it happens, adaptation is an essential ingredient of (long-term) utility, i.e. “systems that . . . allow for change may be suboptimal in the short run but prove wiser in the long run” [16, p. 68]. We believe that this ‘wisdom’ consists of maximising not just utility but also fairness. In this work, by unifying principles of enduring institutions with canons of distributive justice, we argue that

it is open systems which allow for change and fairness that may prove to be wisest in the long run.

ACKNOWLEDGMENTS

The first author was partially supported by QNRF Grant Number 09-1150-2-448. The third author holds a Marie Curie Intra European Fellowship. We are especially grateful for the many helpful comments of the anonymous reviewers.

REFERENCES

- [1] E. Ostrom, *Governing the Commons*. CUP, 1990.
- [2] M. Cox, G. Arnold, and S. Villamayor Tomás, “A review of design principles for community-based natural resource management,” *Ecology and Society*, vol. 15(4), p. 38, 2010.
- [3] N. Rescher, *Distributive Justice*. Bobbs-Merrill, 1966.
- [4] S. Gaechter, “Conditional cooperation: Behavioral regularities from the lab and the field and their policy implications,” School of Economics, University of Nottingham, Discussion Papers 2006-03, 2006.
- [5] A. Jones and M. Sergot, “A formal characterisation of institutionalised power,” *Jnl. IGPL*, vol. 4(3), pp. 427–443, 1996.
- [6] J. Pitt and J. Schaumeier, “Provision and appropriation of common-pool resources without full disclosure,” in *Proceedings PRIMA*, LNCS7455, Springer, 2012.
- [7] S. Brams and A. Taylor, *Fair Division – From cake-cutting to dispute resolution*. CUP, 1996.
- [8] J. Murillo, D. Busquets, J. Dalmau, B. López, V. Munoz, and I. Rodríguez-Roda, “Improving urban wastewater management through an auction-based management of discharges,” *Environ. Modelling & Software*, vol. 26, pp. 689–696, 2011.
- [9] J. Pitt, J. Schaumeier, and A. Artikis, “The axiomatisation of socio-economic principles for self-organising systems,” in *Proceedings SASO*, pp. 138–147. IEEE, 2011.
- [10] A. Artikis, “Dynamic specification of open agent systems,” *Journal of Logic and Computation*, 2011, doi:10.1093/logcom/exr018.
- [11] R. Kowalski and M. Sergot, “A logic-based calculus of events,” *New Generation Computing*, vol. 4, pp. 67–95, 1986.
- [12] J. Rawls, *A Theory of Justice*. Harvard Univ. Press, 1971.
- [13] Y. Chevaleyre, U. Endriss, J. Lang, and N. Maudet, “A short introduction to computational social choice,” in *Proceedings SOFSEM*, LNCS4362, pp. 51–69. Springer, 2007.
- [14] S. Jong and K. Tuyls, “Human-inspired computational fairness,” *JAAMAS*, vol. 22, pp. 103–126, 2011.
- [15] T. Lan, D. Kao, M. Chiang, and A. Sabharwal, “An axiomatic theory of fairness in network resource allocation,” in *Proceedings INFOCOM*, pp. 1343–1351. 2010.
- [16] E. Ostrom and C. Hess, “A framework for analyzing the knowledge commons,” in *Understanding Knowledge as a Commons: From Theory to Practice*, C. Hess and E. Ostrom, Eds. MIT Press, 2006, pp. 41–82.