Laws of Human-Computer Behaviour and Collective Organization

Mark Burgess and Siri Fagernes
Oslo University College, Norway

***Abstract*— We begin with two axioms: that system behaviour is an empirical phenomenon and that organization is a form of behaviour. We derive laws and characterizations of behaviour for generic systems.**

In our view behaviour is not determined by internal mechanisms alone but also by environmental forces. Systems may ‘announce’ their internal expectations by making “promises” about their intended behaviour. We formalize this idea using promise theory to develop an reductionist understanding of how system behaviour and organization emerges from basic rules of interaction.

Starting with the assumption that all system components are autonomous entities, we derive basic laws of influence between them. Organization is then understood as persistent patterns in the trajectories of the system. We show how hierarchical structure emerges from the need to offload the cost of observational calibration: it is not a design requirement for control, rather it begins as an economic imperative which then throttles itself through poor scalability and leads to clustered tree structures, with a trade-off between depth and width.

***Index Terms*— Behaviour, organization, configuration management, peer to peer, pervasive computing.**

I. INTRODUCTION

To manage originally meant ‘to cope’ or to handle[1]. Later ‘manage’ became a transitive verb, i.e. it became something we do to people and systems, like the driving of a vehicle or the running of a business. This is the usage that is most prevalent in computer science today and it has gradually led to a doctrine of control in computer management. This transitive interpretation ignores that fact that a system can only ‘be managed’ or driven if it is both willing and able. The term self-management brings us full circle back to the idea that systems might again cope without external intervention. So the question is again can human-computer systems manage without managers? In this paper, we wish to set aside preconceptions and examine the question in a scientific light.

The organization of functional entities within a system is a key aspect of ‘management’ and is important in delegation and specialization within the resulting phenomena. Organization is therefore closely related to behaviour. In a system of interacting agents, agents can differentiate themselves into different functional roles which work together to extend the scope of their collective behaviour. Regardless of whether such roles and groupings emerge from a dialogue with environmental conditions or are pre-programmed, it is important for an engineer, scientist or analyst to be able to identify these functional elements and understand their relationship to the whole. The term organization is often used for this.

In the natural sciences one does not begin by assuming that observable phenomena can be pre-decided to certain effect. Rather behaviours and phenomena are studied using their causal properties (development in time and space) within the context of an environment (the boundary conditions) to see predictions and outcomes can be discerned from some starting point. This kind of modelling is facilitated by the identification of general laws of behaviour that are stable over time and whose themes recur, leading to a more fundamental understanding of the mechanisms that underlie behaviour. One would like to have such laws in the behaviour of computer systems too.

This paper is about the formulation of such a view within human-computer systems[2]. It is framed in the setting of a theory of maintenance for systems[2] so that we shall take the view that systems can have stable properties even in uncertain environments by arranging for there to be corrective forces maintaining an equilibrium with forces of environmental change.

Specifically this paper is about the relationship between promises made by the parts of a system, i.e. the properties claimed for them and the actions or changes that are required to keep these promises. It links the work of operator maintenence[3], [4], [5], [6] (change management) with the concept of stability and organization. We use promise theory to describe properties and operator mechanics or state machines to describe the kinds of singular and collective behaviour in systems. System components are isolated and modelled as autonomous entities that we call ‘agents’. These should not be assumed to have any relation to the agents in Multi-Agent Systems *a priori*[7].

The structure of our paper is as follows. We begin by mentioning some related attempts to describe organization and behaviour in the literature. In section III we describe the fundamentals of our theoretical framework involving promises and operator algebra. Then we turn to the statement of laws of behaviour in section IV which follow from fundamental ingredients of causality and variation. In section V we sketch the beginnings of an algebra of observation, sufficient to discuss equivalent scenarios and rewriting rules that help to reveal the basis of clustering and cooperation in ensembles of agents. Finally we turn to the existence of patterns of behaviour and the meaning of organization, including its

---

[1] The meaning originates in horsemanship

[2] A shorter preliminary version of this work was presented in [1].

hypothesized economic origin.

II. RELATED WORK

Descriptions of behaviour as an empirical phenomenon are rare in computing. Computer science describes mainly what we shall call programmed behaviour, i.e. that which can be represented by the transitions of a state machine. In software design and network management it is often assumed that systems will behave more or less deterministically, or at least in accordance with our assumptions and specifications. Many unnecessary surprises result from such expectations. In the Unified Modelling Language, for instance, behaviour is represented as algorithms, flow diagrams and state charts. However any system capable of basing its actions on input events from its environment, or whose resources are governed and modified by the same environmental conditions, are necessarily unpredictable from a state machine viewpoint. The situation for distributed systems is often more acute due to the greater exposure to environment of the component parts. Control theory is the paradigm that has come to be used to discuss more advanced feedback behaviour in systems[8].

Policy based management[9] does little to improve on this. The prevailing view is represented by the Event Condition Action (ECA) paradigm which addresses responses to individual environmental stimuli and makes no attempt to consider the long-term consequences of these. Policy based management further defaults to the notion of management by “obligation” and enforcement which we believe is unrealistic[10]. Critics of our views might say ‘what is wrong with enforcement? It’s what we want’. We reply that wanting and having are two different things. We have to confront all uncertainties that make systems hard to control, not ignore or suppress them.

Another difficulty that arises with logic approaches, especially deontic logic, is the confusion that arises between the concepts of “obliged” (assumed to mean “enforced”) and “desired”. In many instances the concept of distributed *coordination* is transformed into *subordination* without further explanation. Promise theory requires this step has to be made explicit, indeed to document how it is possible and whether it is likely to succeed.

Finally, to move from behaviour to organization, one must look hard to find any departure from the idea of hierarchical management. For most researchers the word organization is equated with a hierarchical chain of command. This is true of theories of organization put forward in the business arena, which has been both more innovative and has made considerably more progress than IT management[11], [12]. Coase’s excellent essay took issue with the notion of hierarchy already in the 1930s[11]. More recently the idea of semantic web has been used to try to meld ECA thinking with typed ontologies. Dietz’s view is a theory of actions, typed with the use of formalized ontologies[13]. He introduces the notion of coordination acts, or patterns of transactions. This theory falls under the category of Event Condition Action (ECA) liberally mixed in with some social science.

The work that seems to relate most closely to our comes from the Multi-Agent System community[7], where authors have discussed the concept of “commitment” for many years[14]. It has been suggested that promises and commitments are indeed the same, but there are both philosophical and practical differences to these theories that we mention below.

Ref. [15] discusses how agreement can be achieved through an algorithmic approach to introduction, discharge and withdrawal of commitments. This is somewhat akin to the promise theory description in process algebra by Bergstra and Bethke[16]. This paper, like promises, take pains to deemphasize the role of sequential ordering implied in approaches like UML and BPL.

Ref. [15] considers groups of agents with inhomogeneous capabilities and considers how strategies for coordination can be built. Rather surprisingly, the paper takes the view that subordination is the key to coordination, i.e. the formation of a hierarchy. This is a common view in computer science but the necessity of hierarchy is only asserted, never shown. Ref. [15] declares that commitments are obligations which we find to open a number of semantic confusions. An obligation in the intuitive sense need not be considered a outside directive (with punitive force) or an implied subordination – it need only be a voluntary feeling of motivation to make a voluntary commitment (or a promise).

We find the term commitments difficult to parse due to its duplicity of meanings. To commit to something is an autonomous *action* which implies a binding to some condition or course of action which we do not expect to reverse. A promise is only a declaration of best-effort intent. To commit someone to a course of action (transitive) is to direct actions of a subordinate, which is the opposite of autonomy. The literature on commitments straddles these two meanings, thus the notion of autonomy which is central to the present work is often lost in the multi-agent literature, overshadowed by a focus on programmes of execution. Ref. [17] makes at least clear statements about the formal aspects of commitments to more general ”propositions”; it also considers the matter of conditional commitments which makes it interesting here as conditionals in promise theory are of central importance to its predictive power.

Finally, it has become common to speak of *virtual organizations*, e.g. see [18]. These also define or impose organization based on hierarchical ”management”. The concept of the grid was introduced in the mid-1990s to describe a vision of virtual organizations[19] which follow naturally from distributed resource organization. The Service Oriented Architecture (SOA) has been proposed more recently to describe business motivated sharing of services in a more market oriented framework. These can also be considered as observable cases, but not as theories or definitions.

Today hundreds of papers are written about these topics, but there seems to be little attention to what distinguishes the different terms. How shall we understand these concepts? We base our approach here on promise theory[10], which is well suited to describe the voluntary cooperation of distributed components.

III. PROMISES AND OPERATORS

Promises are a modelling framework (see [10], [20], [21]) which builds from an atomic and fully decentralized view of behaviour in systems. Promise theory describes the persistent features and coordination of "agents" (system components) that are autonomous in the sense of being having private knowledge, and being impervious to outside coercion. Whereas the more traditional notion of obligation leads to distributed constraints[22], promise theory localizes constraints to a single agent through the assumption of strong autonomy.

A promise is the announcement of a fact or a behaviour (commonly expected in the future, but not necessarily) that requires verification to confirm its actuality. Promises last for finite time; they are not events but conditions that persist. A promise is more than an intention, since an intention need not be announced nor event specified to anyone. A promise is different from a commitment, since a commitment is a moment at which an agent breaks with one course of behaviour for another discontinuously with sights on a goal, often through some specific action or investment in the future outcome. In some cases the act of committing can result in a persistent promise as its outcome, but promising does not imply an action that makes a discontinuous change.

Consequences, or results (terms used by other works) are possible outcomes of promises. Indeed, a *goal* in the parlance of many works is now definable as the desired end-point of a promise (see the discussion below). The outcome of a promise is what actually happened, whether a goal was announced or not. We shall discuss outcomes below in terms of trajectories. If one assumes that promises are necessarily kept (the default assumption), this distinction is moot. However, in all realistic systems promises are only kept with a certain probability. This section is about the relationship between promises and and the outcomes of those promise over time. To keep a promise we might have to act or issue an event, maintain a state or prevent change from occurring.

Promises are made by a promiser agent to a promisee agent as a directed relationship labelled with a promise *body* which describes the substance of the promise. A promise with body $+b$ is understood to be a declaration to "give" behaviour from one agent to another (possibly in the manner of a service), while a promise with body $-b$ is a specification of what behaviour will be received, accepted or "used" by one agent from another (see table I). A promise *valuation* $v_i\left(a_j \xrightarrow{b} a_k\right)$ is a subjective interpretation by agent a_i (in a currency of its choice) of the value of the promise in the parentheses. The value can be negative if it is pure cost. Usually an agent can only evaluate promises in which it is involved.

A promise body b has a *type* which describes the nature or subject of the promise, and a *constraint* which explains what restricted subset of the total possible degrees of freedom are being promised. Since any dynamical, systematic behaviour is a balance between degrees of freedom (avenues for change) and constraints[23], this should be sufficient to describe a wide variety of phenomena. For many purposes, and to avoid extraneous concepts, the environment in which agents live and act can itself be represented as an autonomous agent with extensive internal resources. We denote this agent E.

Symbol	Interpretation
$a \xrightarrow{+b} a'$	Promise with body b
$a' \xrightarrow{-b} a$	Promise to accept b
$v_a(a \xrightarrow{b} a')$	The value of promise to a
$v_{a'}(a \xrightarrow{b} a')$	The value of promise to a'

TABLE I
SUMMARY OR PROMISE NOTATION

Promise theory is mainly about the analysis of epochs in which promises are essentially fixed. If basic promises change, we enter a new epoch of the system in which basic behaviours change. For a fixed static set of promises, behaviour continues according to the same basic pattern of interactions between agents and environment.

IV. LAWS OF BEHAVIOUR FOR AUTONOMOUS AGENTS

We expect to find laws of conservation and change in any theory of behaviour. Intuitively one might easily expect the following:

1) An autonomous agent continues with uniform behaviour, unless it accepts an influence from outside.
2) The observable behaviour of an agent is changed when promising to act on input from an outside source (see section IV-B).
3) Every external influence $+b = \langle +\tau, \chi_1 \rangle$ promised by an external agent must be met by an equal and opposite promise $-b = \langle -\tau, \chi_2 \rangle$ in order to effect a change on the agent. If $\chi_1 \neq \chi_2$, then the interaction is of magnitude $\chi_1 \cap \chi_2$.

We shall show that basic laws of this form do indeed apply. In addition, one should expect behavioural properties of any ensemble of agents to be guided by three things:

- The internal properties of the agents themselves.
- The nature of agents' links or bonds (promises).
- The boundary conditions of the environment and location in which the agents evolve.

A. Freedoms and constraints

Definition 1 (Exact and inexact promises): A promise is exact if it allows no residual degrees of freedom. A promise $a_1 \xrightarrow{b} a_2$ is inexact if the constraint $\chi(b)$ has residual degrees of freedom. i.e. if it is not a complete and unambiguous behavioural specification.
For example $q = 5$ is an exact specification, while $1 < q < 5$ is inexact. The same principle applies to the possible outcome of a promise, however the actual outcome is naturally exact in each measurement.

B. Behavioural trajectories

To discuss behaviour over time we need to notion of a *trajectory*. This is the path taken by (i.e. the set of intermediate states between the start and the current value of) an agent's observables through a space of states that we may call configuration space. It represents the past or future history of an agent's state transitions. Let $\vec{q}$ be a vector of state information (which might include position, internal registers and other details)[2]. Such a trajectory begins at a certain time t_0 with a certain coordinate value $\vec{q}_0$, known as the *initial conditions*.

The trajectory of a single agent is then a parameterized function $\vec{q}(t, \vec{\sigma})$, for some vector of parameters $\vec{\sigma}$ arriving from an outside source, and we identify the behaviour of an isolated system as the triplet as the determined trajectory:

$$\langle \vec{q}_0, t_0, \hat{O}(\vec{\sigma}) \rangle, \; t > t_0. \tag{1}$$

The symbol $\hat{O}(\vec{\sigma})$ is a constant transition matrix or operator which takes $q(t_i)$ to $q(t_{i+1})$ for integer time index i, or alternatively $q(t)$ to $q(t + dt)$ in a differential form. We can think of this operator as being the generator of time slices, advancing by one time step on each operation; $\hat{O}(\vec{\sigma})$ therefore represents a steady state behaviour and any alteration to this steady state behaviour must come about by a transformation $\hat{O} \rightarrow \hat{O}'$, which by the rules of algebraic invariance must have the form $\hat{O}' = T^\dagger \hat{O} T$ for some matrix T and dual-transpose representation †[3].

In other words, any change in an agent's state (called its behaviour) is generated by

$$\vec{q} \rightarrow \vec{q}' = \vec{q} + \delta\vec{q} = \hat{O}(\vec{\sigma})\vec{q} = (1 + \hat{G}(\vec{\sigma}))\vec{q}. \tag{2}$$

i.e. $\delta\vec{q} = \hat{G}(\vec{\sigma})\vec{q}$. $\hat{G}(\sigma)$ is called the generator of the transition $\hat{O}$; $\delta\vec{q}$ plays the role of a generalized momentum or 'velocity', so that the dynamics state is represented by the canonical pair $(\vec{q}, \delta\vec{q})$.

We now have a simple transition matrix (or state machine) formalism for describing the steady state behaviour of an agent, which results from keeping its promises through the repeated action of a 'promise keeping operator' $\hat{O}$. An agent whose observable properties do not depend on any external circumstances has *exact* or *rigid* rigid behaviour[24], [25]. It is possible if and only if the agent has no use-promises that pertain to its own behaviour ($-b$ for some b), and all other promises $+b'$ are exact promises. In this case the internal change operator $\hat{O}$ cannot depend on any external information.

C. Outcomes and goals

The notion of a trajectory as a representation of behaviour allows us to be more precise about the meanings of other commonly used terms. We define the *collective behaviour* of several agents simply as the bundle (i.e. direct sum) of trajectories of the ensemble of agents.

An *outcome* (which can equally well refer to the outcome of a promise or of a transition taken to keep a promise) can be described as a single point $q(t_{\rm final})$ of the configuration space reached at some 'final' time, along the trajectory of an agent. In other words it is an identifiable end-point of an agent's behaviour.

A *goal* is then a set of one or more desired or acceptable outcomes within an agent's own state space. In other words a goal is a bounded space-time region that the agent would like its trajectory to intersect (like the bull's-eye of a target). A set of $\{q(t)\}$ possibly over some region $t_{\rm min} \leq t \leq t_{\rm max}$. This definition obeys the principle of autonomy, namely that an agent may only promise its own behaviour; however it leaves open the question of how an agent might desire to change its environment (which is outside of its own state space). We come back to this point in section IX since it requires the notion of force.

We wish to point out that a goal cannot be an elementary concept like the subject of a promise, since it requires a feedback loop to achieve, which requires several promises. A goal requires knowledge of the state to be reached and therefore the ability to observe the state and its current state to know when intersection has occurred. As long as the state is internal to the agent we can assume it can simply make these promises itself. However, the situation is much more complex where multiple agents are involved. A collective goal requires all agents to achieve a pre-arranged goal simultaneously. This requires not merely private promises but coordination and hence multiple two-way communication between the agents.

Notice also that the concept of a goal requires the notion of a value-judgement about what is desirable or acceptable. This is easily provided in the promise framework if we always refer to the outcomes of promises, but again it is highly complex where multiple agents are involved.

Finally we should at least mention the notion of non-deterministic states, i.e. macro-states in which a goal is achieved only on average over some interval of time. A promise, after all, lasts for some time and is verified perhaps several times. A promise therefore leads to a distribution of outcomes in general, not merely a single state. One may thus define an *equilibrium* as a goal that is satisfied by a stable distribution over a 'sufficiently persistent interval of time'. As this raises many questions to be answered about the statistical mechanics of agents, we shall defer a full discussion of statistical behaviour for later work.

D. Changes to steady state

Now, consider how an agent might exhibit behaviour that is based on input from another agent. To see how we might affect a change in this behaviour generated by $\hat{O}$ we need to follow the straightforward rules of matrix transformations. Reactive or adaptive behaviour means that autonomous agents make promises to accept input from an external agent. Thus the operator must be made functionally dependent on the input $\hat{O} \rightarrow \hat{O}(I)$. This requires a promise binding to accept input conditionally on its provision, e.g.:

$$a \quad \overset{+O(I)/I}{\rightarrow} \quad a'_{\rm ext} \tag{3}$$

$$a \quad \overset{-I}{\rightarrow} \quad a_{\rm ext} \tag{4}$$

where I represents a promise of input from an external agent, and $O(I)$ represents a promise of some observable output to another external agent which is conditionally a function of the input, and is kept via the operation of $\hat{O}(I)$.

---

[3] This is a linear transformation. It is not certain that all transformations of the operator need be linear, but we make this assumption here for later work to extend. Ref. [3] shows examples satisfying such linearity, and our results here are only for linear mechanics.

Let $I \rightarrow +\Sigma_\tau$ be the body of a promise to change the generator of behaviour $\hat{O}$ from an external agent: i.e. $a_{\rm ext} \overset{+\Sigma_\tau}{\rightarrow} a_i$. The agent whose behavioural generator $\hat{G}$ is being altered promises to accept the change with $a_i \overset{\cdot\Sigma_\tau}{\rightarrow} a_{\rm ext}$ and we denote a linear realization of the operator which keeps the promise to use this transformation by the external agent simply by Σ_τ so that we have:

$$\hat{G}_\tau \rightarrow \hat{G}' = \Sigma_\tau^\dagger \hat{G}_\tau \Sigma_\tau. \tag{5}$$

The generator of this transformation matrix can, in the usual way, be written as σ_τ where $\Sigma_\tau = I + \sigma_\tau$, and

$$\delta\hat{G}_\tau = \hat{G}' - \hat{G} = \sigma_\tau^\dagger \hat{G} + \hat{G}\sigma_\tau + \sigma_\tau^\dagger \hat{G}\sigma_\tau \tag{6}$$

What can we say about the transformation matrix? In order to satisfy the principle of autonomy it must have the following properties. Let us define a valuation of a promise known as the *outcome* by the notation: $o(a_1 \overset{b}{\rightarrow} a_2)$. The outcome returns a value in $[0,1]$ where 0 means not-kept and 1 means kept. Intermediate values can be used for any purpose, such as statistical compliance. Autonomy requires us to stipulate:

$$\begin{aligned} \Sigma_\tau &\propto o(a \overset{-\Sigma}{\rightarrow} a_{\rm ext}) \\ \Sigma_\tau^\dagger &\propto o(a \overset{+\Sigma}{\rightarrow} a_{\rm ext}) \end{aligned} \tag{7}$$

so that

$$\begin{aligned} \delta G &\rightarrow 0 \\ \Sigma_\tau &\rightarrow 1,\ \Sigma_\tau^\dagger \rightarrow 1 \\ \sigma_\tau &\rightarrow 0,\ \sigma_\tau^\dagger \rightarrow 0, \end{aligned} \tag{8}$$

when one of the binding promises with the external agent is not kept. This means that, unless the promises to deliver an interaction influence are honoured by both parties, then steady state behaviour persists.

The above boundary conditions are the only interpretation of interaction that preserve the requirements of autonomy.

E. Laws of change

We now state the basic law of causation for behaviour in terms of the autonomous promises of the agents, under the condition of autonomy.

Law 1 (Law of Inertia): An agent's observable properties hold a constant, deterministic trajectory $\vec{q}(t)$ unless it also promises to use the value of an external source $\vec{\sigma}$ to modify its transition matrix $\hat{O}(\vec{\sigma})$.

Proof: This follows from eqn. (6). Steady state trajectories imply that $\delta G = 0$, which in turn requires that for small changes $\sigma^T = 0$, which implies no promise of type $-\Sigma$. ■

Put another way, each agent has access only to information promised to it, or already internal to it. A local promise $a_i \overset{f(\vec{\sigma})}{\rightarrow} a_j$ that depends on an externally promised parameter σ is clearly a conditional promise $a_i \overset{f(\vec{\sigma})/\vec{\sigma}}{\rightarrow} a_j$, where $\vec{\sigma}$ is the value promised by another agent. In order to acquire the value of $\vec{\sigma}$, we require $a_i \overset{-\vec{\sigma}}{\rightarrow} a_j$ and a corresponding promise to provide $\vec{\sigma}$ to a_i either from the environment or from another agent. Thus, if an agent does not promise to use any input $\vec{\sigma}$ from another agent, all of its internal variables and transition matrices must be constant.

Note also that by the definitions in [20], a conditional promise is only a promise when combined with a use-promise. This fits naturally with the argument in the theorem.

Corollary 1 (A conditional promise is not exact): By reversing the theorem we see that a conditional promise must, by definition have a residual degree of freedom, which is the value of the dependent condition.

We can now state the interaction mechanics using the formulations of the previous laws, and in terms of clear statements about state transitions:

Law 2 (Law of interaction): The acceleration $\delta^2 q$ of an agent's promise trajectory resulting from a promise $a \overset{O/I}{\rightarrow} a'$ (i.e. the rate of change of its generalized momentum δq) is proportional to the generalized force $F = \delta\hat{O} = \delta\hat{G}$ promised by an external agent.

Proof: This now follows trivially from the transformation properties and boundary conditions:

$$\delta_\tau \vec{q} = \hat{G}_\tau \vec{q} \tag{9}$$

where $\hat{G}_\tau$ is the matrix valued generator of behaviours of type τ, see eqn. (2). Under a change of G

$$\begin{aligned} \delta\vec{q} &= \hat{G}\vec{q} \\ \delta\vec{q}' &= \hat{G}'\vec{q} \end{aligned} \tag{10}$$

thus

$$\delta^2\vec{q} = \delta\vec{q}' - \delta\vec{q} = (\hat{G}' - \hat{G})\vec{q} = \delta\hat{G}\,\vec{q}. \tag{11}$$

■

Law 3 (Transmitted force - reaction to influence): The effective transmitted force due to a promise binding between two agents is that which results from the outcome of the body-intersection of equal but opposite (±) promises between the agents.

Proof: By the assumption of autonomy, the influence of agent a by $a_{\rm ext}$ is the conjunction of information sent and information accepted: $influence = offer \wedge acceptance$. This has an obvious set theoretic formulation[20]. From the rules of promise composition, the binding

$$\begin{aligned} &a_{\rm ext} \overset{+\langle\tau,\chi_1\rangle}{\rightarrow} a \\ &a \overset{-\langle\tau,\chi_2\rangle}{\rightarrow} a_{\rm ext} \end{aligned} \tag{12}$$

has an outcome that satisfies:

$$\begin{aligned} &o\left(a_{\rm ext} \overset{+\langle\tau,\chi_1\rangle}{\rightarrow} a, a \overset{-\langle\tau,\chi_2\rangle}{\rightarrow} a_{\rm ext}\right) = \\ &o\left(a_{\rm ext} \overset{+\langle\tau,\chi_1\cap\chi_2\rangle}{\rightarrow} a\right) o\left(a \overset{-\langle\tau,\chi_1\cap\chi_2\rangle}{\rightarrow} a_{\rm ext}\right), \end{aligned} \tag{13}$$

so that the interaction is the intersection of the agents' promises to give and receive the influence. ■

Thus we can say that the trajectory's transformation must have the form:

$$\underbrace{\delta\hat{O}}_{\text{Force}} \simeq \underbrace{o(a_{\rm ext} \overset{+\Sigma}{\rightarrow} a)}_{\text{field}} \underbrace{o(a \overset{-\Sigma^\dagger}{\rightarrow} a_{\rm ext})}_{\text{charge}} \tag{14}$$

There is a reassuring correspondence here with the physics of force fields, which is a directly analogous construct, where ± charge also labels which particles promise to respond to one another's field. Promises appear like fields of influence whose values are sampled by the action of measurement. Promised behaviour is represented by the regular application of operators $\hat{O}$ on a state vector that evolve the state and keep the promise. The outcome is unknown until the act of verification is initiated, somewhat analogous to the quantum theory of matter.

We emphasize that these laws derive directly from the assumptions of autonomy, operational change and transition matrix formulation of the agents. They are therefore beyond dispute and we would expect to find this kind of law in any system of change with similar properties.

V. OBSERVATION AND MEASUREMENT

Measurement and observation are central to the discussion of system behaviour. The concepts of calibration and coordination must therefore also feature. Promises define the only observables in promise theory. No knowledge is predictably exchanged without it being promised, so there is no measurable certainty without both promises to be observable and to observe in a binding relationship. For instance, the location and nature of an agent can only be observed if the agent promises to make itself visible. This includes interactions with an environment. The environment itself must promise its secrets to agents. Although this is clearly a device, this mode of description has the advantage of making explicit all assumptions about individuals and their interactions, including observations of environment and boundary conditions etc.

The strong assumption of autonomy in promise theory underlines an essential limitation on such interactions between agents. Agents observe according to their own private standards, thus each agent makes individual valuations of what it observes. This might not be the same as other agents (indeed no agent can know about another agent's perception of the world; at best they can try to communicate their own experiences and seek consistency). Only relative comparisons can therefore be expected to have meaning.

This limitation suggests a basic algebra of measurement because it implies that all comparisons must be made through a single adjudicator. Indeed this property explains how certain agents attain a privileged position in an ensemble, by having access to information and using a single scale of measurement for coordination of all the information from observed agents.

We use the shorthand notation $C(b)$ for a pattern of promises that leads to the effective promise "exhibit the same as". The coordination promise is used for this (see fig. 1). The reduction rule for coordination promises for the case in which n_1 promises n_2 that it will coordinate on the matter of b, given that n_2 promises n_3 b follows. The symbol "⊗" is used to signify the composition of these promises.

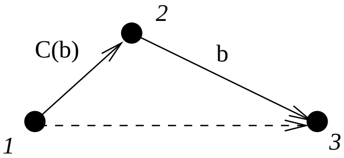


Fig. 1. Serial composition of a promise and a coordination promise. The dashed arrow is implied by the $C(b)$ promise.

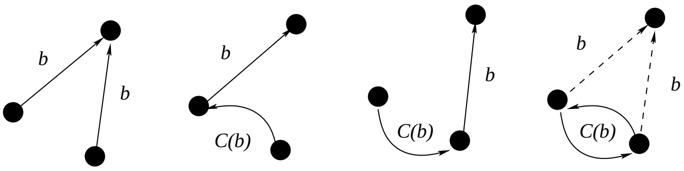


Fig. 2. Observation indistinguishability implies an equivalence.

$$\underbrace{n_1 \xrightarrow{C(b)} n_2}_{\text{'Coordinate with'}} \otimes \underbrace{n_2 \xrightarrow{b} n_3}_{\text{Promise}} \Rightarrow n_1 \xrightarrow{b} n_3 \; . \tag{15}$$

The coordination promise is transitive.

$$n_1 \overset{C(b)}{\rightarrow} n_2 \, , \, n_2 \overset{C(b)}{\rightarrow} n_3 \Rightarrow n_1 \overset{C(b)}{\rightarrow} n_3. \tag{16}$$

We use this below in the identification of observable properties, since it implies a basis for n_3 to compare n_1 and n_2.

A. Distinguishability

It follows from the discussion surrounding the third law that the outcome perceived by agents a_1 and a_2 in their observation of a third agent a_3 need not agree. Their promises to receive promised data could be different or even incorrectly calibrated. This is not the only reason why perceived values might differ, but it is a sufficient one. It follows that each agent has an independent estimation of values observed. How then can any third agent determine whether a pair of agents has the same behaviour? To do this measurement must be with respect to an arbitrary but *singular* observer that can make relative comparisons according to its own subjective apparati.

The coordination rewriting rule can be applied both forwards and backwards. Since an agent that observes behaviour b has no knowledge of what might lie behind it, it cannot tell the difference between the scenarios in fig 2.

Since agents cannot distinguish between these cases by observation alone, they are entitled to consider the situations equivalent.

Definition 2 (Equivalence under observation): A constellation of two or more agents that promise identical observables with equivalent behaviour to a given agent are considered equivalent under observation if the observing agent cannot distinguish the average behaviour of the agent with respect to the promised observable.

Note that equivalence could be exact at any moment in time, perfectly synchronized changes, or it could be defined as an equivalence of averages over a sampling interval (e.g. a suitable time scale). This must be specified when reasoning about the promises.

The notion of equivalent is not entirely clear however.

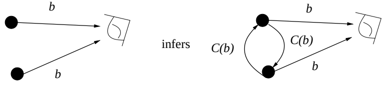
- Equivalent means identical.
- Equivalent only statistically, over a given sample scale, with quantifiable bounds.

These judgements depend clearly on the abilities of the observer to discern and distinguish behaviour, thus two different agents' findings should cannot be compared without prior calibration.

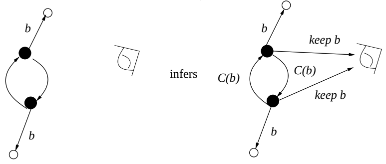
B. *Rewriting for cooperative behaviour*

Based on the equivalences shown in fig. 2 a number of rewriting rules can be formulated expressing equivalences under our observation as modellers with complete information about all agents (having a globally privileged insight that no single agent has). We shall describe them pictorially here to avoid unnecessary formalism. From the symmetry of indistinguishable agents, the implicit promises between the agents must appear in both directions.

1) *Inferring implicit coordination.* Two agents that behave in the same way to a real or fictitious external observer over some defined timescale can be assumed to be coordinated, and we may introduce symmetrical coordination promises.



2) A corollary to this is that when all agents in an ensemble make identical promises to each other in a complete graph, we can add mutual coordination promises between all pairs. This is easily justified as it reflects the observation that when a number of agents behaves similarly with no labels that otherwise distinguish special roles, an external observer can only say that all of the agents are behaving in a coordinated way. Thus the observer sees a coordinated group for all intents and purposes, although it was not formally agreed by the agents.
3) *Inferring observational indistinguishability.* Any two agents that mutually coordinate their behaviour may be considered to behave analogously over the sampling interval to a hypothetical external observer. This can be formulated by introducing a fictitious promise to the fictitious observer.



We have argued that these rules are the basis for understanding swarm behaviour[26].

Coordination of agents could of course be arranged by having each agent subordinate itself to the instructions of a third party, but in this case one must postulate the existence of an additional *real* agent which does not seem justified directly. However, by combining the rules above one can postulate the existence of such an agent entirely on the basis of equivalences, which is preferable.

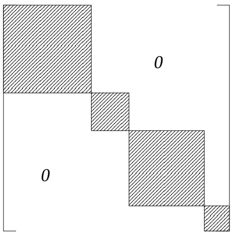


Fig. 3. Patterns in agent space arise from irreducibility of the graph.

VI. PATTERNS OF BEHAVIOUR

To summarize the foregoing sections, behaviour is a pattern of observed change in the observable measures of an agent. When several agents are involved, we speak of collective behaviour. Behaviour is governed by the interplay between degrees of freedom and constraints[23]. In promise theory, changes to observables are assumed to occur through the action of operations[27]. Actions generate events whereas promises are usually persistent claims about likely distributions of outcomes. Collective behaviour refers to a collection of agents that together form a behavioural pattern.

In studying behaviour we are interested not only in singular events but in their trends and classification. This is where promises play a role: promises are long-term and change only adiabatically compared to the events which test them. To classify and analyze such events into a picture of behaviour we must understand their variability.

A. *Patterns*

Patterns can arise in two dimensions:

- Serial patterns of events in time.
- Spatial patterns of relatedness arising from the agents' promise ties. Clustering in the space of promise types - agents with similar promises are expected to behave similarly

Discrete patterns are described by the Chomsky hierarchy of grammars[28]. We may consider each distinct promise, or the operator that transmutes it into a persistent state to be a symbol in an alphabet Σ. The strings formed from this alphabet represent all possible behaviours in time, i.e. all sequences. The matrix of all such strings whose rows and columns represent the agents in an ensemble is also a matrix (a matrix of matrices) of some dimension. Patterns formed by reducible and irreducible blocks along the diagonal of this matrix represent patterns in the space of the agents (see fig. 3).

B. *Order and disorder*

We can characterize the variability of the trajectories resulting from promises.

- *Ordered behaviour:* An agent whose observable properties change according to a deterministic algorithmic pattern with a predictable grammar.

- *Disordered ("random") behaviour:* An agent whose behaviour changes in an unpredictable manner.

In general this is not a binary choice but a continuously varying scale, which is most naturally defined in terms of the entropy of the trajectory. We can define the order of an ensemble by

$$\mathrm{Order} = \left(1 - \frac{S}{S_{\max}}\right) \tag{17}$$

where S is the Shannon entropy of a trajectory defined by

$$S = -\sum_i p(\hat{O}_i) \log p(\hat{O}_i), \tag{18}$$

and $p(\hat{O}_i)$ is the probability or normalized frequency of operator type i occurring in the time evolution of the behaviour.

C. Roles

Roles are labels for agents derived from their observed behaviour. The roles played by agents in an ensemble can be assigned simply by looking for all repeated patterns of promises. A role is then a pattern. Since similar promises will lead to similar observed behaviour, so one defines roles for each distinct combination of promises that occurs in a promise graph. Thus one finds:

- *Differentiated behaviour:*
  Agents that behave differently, e.g. perhaps partitioned into a division of labour when cooperating, or simple independent.
- *Undifferentiated behaviour:*
  Agents play identical roles in the ensemble and require no specific labels, since all promises are made by each agent.

Undifferentiated behaviour can be coincidental i.e. uncalibrated, like a disordered gaseous phase of matter (all the component elements make identical unconditional promises but never interact with one another) and it could imply agents that have agreed to behave alike through interaction (like in a solid phase of matter). Normally we are only interested in the possibility of coordinated, collective phenomena, but a phase transition from one to the other is possible and we shall describe this elsewhere.

D. Formation of hierarchy

Hierarchy lies behind practically all visions of what constitutes systematic and organized behaviour in the literature. We propose in line with many previous authors' thinking that hierarchies emerge for microeconomic reasons. Each agent pursues its own interests selfishly and the resulting collective behaviour reflects an evolutionary process[29], [30]. We want to resist the *assumption* of hierarchy but we cannot deny its widespread dominance empirically.

The characteristic of hierarchy is the existence of a root node, or privileged agent at the top. the question is how this node gets selected from a group of agents. A full understanding of this phenomenon requires a discussion of symmetry-breaking, which is beyond the scope of the present paper, however we can discuss a simplified view. We suggest that this emergence of privilege has a simple explanation in a process of structural 'crystallization' which is seeded by a self-appointed promiser of a collation service. The economic advantage to the leaves for a privileged observer is that can make comparisons within the initially flat ensemble more cheaply than having each individual agent establish peer-to-peer communication with every other agent in the ensemble. The cost benefit of such centralization depends on how many promises need to be set up and maintained.

There are two separate economic issues in ensembles: the cost of *calibration* or the attainment of global measures allowing consistency, and the cost of *coordination* or differentiation and delegation which requires only local consistency. Calibration requires complete bi-directional communication between all agents. This is a familiar problem in security where it is used for key distribution[31]. Coordination requires only that we can pass a message to every agent on a need to know basis, without the necessity for reply. Without calibration, agents have only local concerns and global ones are considered to 'emerge' i.e. they are un-calibrated (we return to emergent behaviour below).

The local economics of network relationships are quite simple and depend mainly of the topology at a point. We need to show how the cost of a particular topology impinges on the cost of either coordination or calibration. We should recall that promises are not about continuous network communications, so the cost of making a promise is entirely in the establishment of the promises. The maintenance of the promise depends on its type however. Promises that require an exchange of information between agents involve propagation of data, which introduces measures of time-taken, latency etc.

Promise graphs form networks and the economics of coordination thus have two facets (see fig 4): cost and efficiency. The cost of establishing promises increases with the number of promises since each promise generally requires some behaviour or work to be done by the agent. The efficiency of coordination involves communication and therefore has to do with propagation of effect over the coordination distance (this is a network depth issue). We can divide the discussion into what is good for the group and what is good for the individual agent.

E. Global considerations

There are two extreme cases for topological connectivity in a global region and a range of values in between. These are the complete graph (all pairs nodes linked peer to peer with $N(N-1)$ directed links) and that of centralized hub (with $(N-1)$ nodes linked directly to a single hub, making $2(N-1)$ directed links). If we assume that, to a first approximation, agents are homogeneous and value promises from one another equally then the cost and value of promises is proportional to the number of promises.

If agents do not use route messages for each other (requiring many coordination of promises), they have to coordinate with every other agent individually in a complete graph of $N(N-1)$ promises of each type of promise in the ensemble of size N.

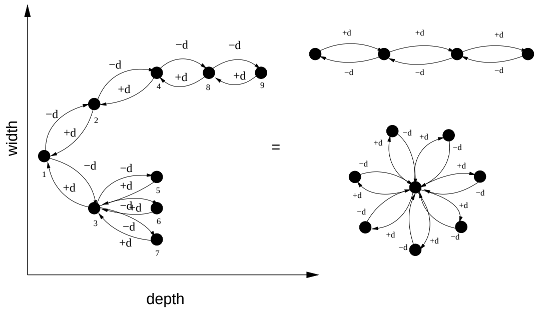


Fig. 4. Bilateral communication structures indicating “depth” and “width” of promise bindings. A tree is something between the extremes of a chain and a hub.

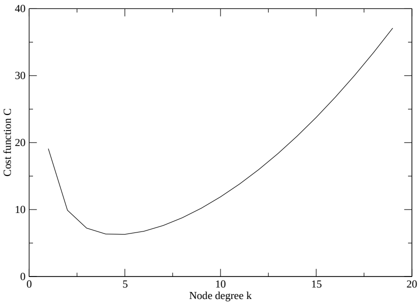


Fig. 5. Cost considerations can plausibly lead to an optimum depth of network pattern when power considerations are taken into account. The minimum cost here is given for $k = 5$. Such considerations require an arbitrary choice to be made about relative importance of factors.

If they network their efforts however into a hub or chain then they can reduce their promises to order $2(N-1)$ in total, but now there is a new issue: depth or efficiency.

Depth versus width is a trade-off. Greater centralization reduces depth and hence increases the coordination efficiency, but it increases the cost burden of promises at the hub. The costs are in-homogeneously distributed. In a chain (the opposite of a hub), the cost of keeping promises is maximally distributed but the depth is maximal too, meaning low coordination efficiency and delays.

F. Local considerations

Agents do not generally see the global picture; they care only about their own costs and benefits. This means that the true picture of cooperative behaviour will necessarily be inhomogeneous in all cases but a complete graph, which all agents will perceive to be expensive for large N. The difference between these must be $N(N-1) - 2(N-1) = (N-1)(N-2) \gg 0$ for justify appointment of a privileged collator. As soon as the privileged collator has been chosen, the cost to non-privileged agents is simply 2. However, as N grows, the cost for the appointed collator grows linearly like $2(N-1)$. One solution is to look for a balanced tree, like a Cayley tree[32] which allow constant scaling of promise cost for all agents, however in this case the depth of the structure increases, leading to a rapid fall-off in efficiency, thus there is a trade-off.

Optimizing the structure is a simple matter of comparing the relative economic merits of these two properties. Let k be the average node degree for promises in a tree. The cost associated with not getting data quickly is proportional to the effective depth of the network pattern $(N-1)/k$, then we have a cost function that is a balance between these two. All cost functions contain arbitrary (subjective) parameters. In this case we denote ours α:

$$\text{Cost} \propto v_i \left(\bigoplus_i (a_i \xrightarrow{-d_i} a_{i-1}) \right) = \alpha \left(k_i^{(-d)} \right)^2 + \frac{(N-1)}{k_i^{(-d)}}. \quad (19)$$

A plot for this for the arbitrary policy $\alpha = 0.1$ is shown below. This shows the existence of an optimum aggregation degree, in this example $k = 5$. Such arguments should also be taken into account in the scaling argument, as we see the cost rise sharply with increasing centralization. This shows a plausible explanation for why a hierarchy emerges. It seems locally cheaper per agent than a full mesh and it can tune its efficiency as long as the structure does not become fixed.

There is a paradox here: agents break symmetry to appoint a leader in order to cheaply scale the number of interactions required to compare and calibrate outcomes from multiple agents for the “client” agents, however the appointed agent ends up choking on this burden eventually. The solution that generally emerges is a kind of lazy-evaluation: agents do not make promises that they do not need to make. What then emerges is often something like a small-worlds network or power-law structure[33], [34], [35] which is seen in peer to peer networks. This suggests that ordered management is not something that scales without active abstention from coordination.

VII. ORGANIZATION FROM DIFFERENTIATION

Let us now examine the word ‘organization’ and try to provide a definition below that is unambiguous. How does *organization* differ from *order*, for instance? In natural science, self-organization generally means spontaneous differentiation or clustering, i.e. a reduction in local entropy of an open system.

Is a tree considered organized, or merely ordered? The now established term self-organization forces us to define the meaning of organization clearly, since it implies that organization may be something that is both identified *a priori* by design, or *a postiori* as a system property.

Intuitively we think of organization to mean the tidy deployment of resources into a structural pattern. “Organization” (from the Greek word for tool or instrument) implies to us a tidy compartmentalization of functions. We know that all discrete combinatoric patterns are classified by grammars of the Chomsky hierarchy[28], [23], which may be formed from the alphabet of such operators. This is consistent with the concept of lowered entropy and differentiation. Organization requires distinguishability.

Patterns may be formed over different degrees of freedom; for instance:

- Spatial or role-based partitioning of operations between parallel agents.

- Temporal (schedule), i.e. serial ordering of operations at an agent.

For some, an organization also imbues a conscious decision amongst a number of agents to work together, with a hierarchical structure, and a leader, e.g. with a *separation of concerns*, or *division of labour* in the solution of a *task*. Many also believe in the value of *re-usability* (a subjective valuation of implementation which could lead to an economic criterion for selection of one structure over another).

We prefer to think that all of these can be understood economically. Two agents trained to fight a fire could both independently promise to grab the fire extinguisher or dial 911, but if they promise to divide the tasks then both tasks will be started sooner and finished earlier costing less totally and improving efficiency by parallelism.

Parallel efficiency gain is the seed for *differentiation*; its survival is a matter of sustained advantage, which requires sustained environmental conditions.

We define an organization as a discrete pattern that is formed from interacting agents and which facilitates the achievement of a desired trajectory or task, i.e. a change from an initial state $\vec{\mathcal{Q}}_{\rm i}$ to a final state $\vec{\mathcal{Q}}_{\rm f}$ over a certain span of time. We refer to the discussion of systems in ref. [2] for the definition of a task.

Definition 3 (Organization): A phenomenon in which a pattern forms in the behaviour of an ensemble of differentiated agents.

Let $\mathcal{E}$ be an ensemble of distinguishable agents. The observables of $\mathcal{E}$ can formally be written by direct sum $\vec{\mathcal{Q}} = \vec{q}_1 \oplus \vec{q}_2 \oplus \ldots \vec{q}_N$, but we do not assume that these are public knowledge to an actual agent. An organization over the ensemble consists of the tuplet $\mathcal{Z} = \langle \mathcal{E}, \mathcal{Q}, \mathcal{A}, \mathcal{S} \rangle$, where: $\mathcal{E}$ is a set of agents with a promise graph $\mathcal{A}_{ij}$. $\mathcal{S}$ is a string of matrix operators for the whole ensemble $\hat{\mathcal{O}}_A(a_i \stackrel{\pm *}{\rightarrow} a_j)$ which describes the observable changes made by agents, for some sequence index A, Diagonal elements of $\mathcal{S}$ include the operations $\hat{O}_A(t, \vec{\sigma})$. $\mathcal{S}$ spans all the observables in the ensemble with column dimension $\sum_{i \in \mathcal{E}} \; \mathrm{dim}(\vec{q}_i)$, and modifies the observables of all agents: $\hat{\mathcal{O}} \vec{\mathcal{Q}} = \vec{\mathcal{Q}}'$.

Organization can now be understood as a discrete pattern induced with $\mathcal{Z}$. We discern orthogonal types of organization (analogous to the longitudinal and transverse nature of wave patterns in a continuum description):

- **Serial organization** is the syntax of operational changes $\mathcal{S}$, classified by a Chomsky grammar.
- **Parallel organization**: is the partitioning of $\mathcal{Q}$ induced by the irreducible modules of $\hat{\mathcal{O}}_A$ at each serial step. This is a property of multiple agents and is characterized by the eigenstructure of $\hat{\mathcal{O}}_A$, which defines natural regions in the graph[36].

What of 'an organization' as a noun (e.g. an institution or company)? We normally think of this as a number of actors who are organized under the umbrella of some architectural edifice – like a building. Is it enough to simply collect agents within a boundary to make them an organization? We think not. Our definition above still works in this case, but it does not quite fit the facts. An organization is clearly an ensemble with collective behaviour, and it clearly forms a pattern (even a trivial one); however, organizations or institutions as we understand them always have boundaries (which one may or may not consider artificial).

The only natural boundary for interaction is to limit our understanding of organization to the point where no more promises are made. However, this would mean that every business had all of its clients as part of its organization, which is not our common understanding of what an organization is. Where is the edge of a pattern? The resolution lies in the common usage of organization as a synonym for institution. If an organization could have a boundary it would be completely isolated from other agents, a breach in a boundary (a leaky boundary in the parlance above) would demand that we extend the boundary to include the part it is interacting with. This in turn means that the boundary is not a boundary. Any ad hoc definition of the edge of the organization (the edge of the pattern) would be arbitrary and subjective, and no agent would be able to know whether it were inside or outside the organization itself. However, therein lies the clue. How would an agent know? The boundary is not defined by interaction but by a specific promise type, by a promise of membership. The common meaning of an organization is as follows:

Definition 4 ('An' organization): A number of agents that each promises to be identified as members of an organization.

Organization is thus more than a pattern that identifies 'collective agents' making promises that a single agent would not be able to make alone. They are also self-appointed *roles*.

VIII. EQUILIBRIUM

When the outcome of one agent or organization is promised to another agent or organization and vice versa, and the result is a pair of persistent trajectories, we refer to the relationship as economic trade. The phenomenon is called symbiosis in biology. This mutual closure between promises is a basic topological configuration that allows the persistence of an operational relationship (an ecosystem). When the trade of promises is stable over some time, the result is a dynamic *equilibrium*.

Equilibrium does not imply static fixture. Dynamic or statistical equilibria describe properties that are in balance on average. This is the more normal state of affairs, since noise from the environment can never be completely shielded.

A slow changes in the properties of the agents or the environment can lead to a drift in the average values. If this drift is slow enough to be distinguished from the fluctuations themselves then we may call it *adiabatic*. This means simply that there is weak enough coupling between the fluctuation process and the process leading to average drift to lead to a clean separation of scales. Systems that have exclusively strong coupling do not exhibit this property and are much less predictable as a result[37], [38]. The interaction of scales is a vast topic that cannot be given a fair treatment here. Suffice it to say that this is a crucial part of behavourial description in any system and the promise binding description allows us to understand this in a classic interaction viewpoint.

Our formulation of agents interacting through persistent promises reveals equilibria more clearly than a description in terms of events and actions would be able to, because it approaches the problem from the scale of the persistent phenomenon.

IX. AGENTS, ENVIRONMENT AND EXTERNAL GOALS

By the first law, systems are most predictable when completely isolated from external forces. At the next level, their changes can be predicted when coupling to external forces is weak. The stronger the coupling between agents, the more unknown information enters each agent. This can lead to disordered behaviour which requires more information than is practical or available to understand.

In promise theory all agents begin by default in a state of isolation, impervious to outside influence. It is only through their own promises that they can volunteer to be influenced.

Three questions remain in the discussion: i) how do we explain irresistible forces such as weather, power-failures and other ‘acts of god’? ii) How do we model the fact that an agent can affect its environment, e.g. draw graffiti, move an object etc? Finally, iii) how do we model the presence of boundary conditions, or restrictions over which agents have no choice? The concept of force is similar to that of an attack:

Definition 5 (Attack/Force): An attempt to alter an agent’s trajectory without its consent (i.e. in the absence of a use-promise). This is a breach of autonomy.

Let us consider these briefly for completeness, but defer a full discussion for later work. There are two classical approaches one might take to modelling environmental forces. The first is to think of the environment as simply one or more surrounding agents that distinguish themselves by the magnitude of their influence. The alternative is to treat external objects including the system boundary as being “something else”, i.e. some kind of external object that is not an agent. To justify the latter approach we would have to extend the framework of this paper to say what we mean by such an external force and thus we avoid this in the present work. We wish instead to give a very simple view of environmental interaction by treating the environment as a single “super-agent” which promises to allow itself to be changed by any agent and to which all agents have “voluntarily” promised to be influenced. Although this is somewhat artificial[4], it allows us to continue our simply formalism without unnecessary complications.

How can an agent move an object in promise theory? The state of the object needs to be represented in a state space and we must be able to discern its trajectory. If the object is of sufficient importance we can model it as a separate agent that promises to allow itself to be moved by another. Alternatively we can consider all such objects to be mapped into the state space of an environmental super-agent. This super agent can be influenced and can influence agents.

Definition 6 (Leaky agents): We define a *leaky agent* to be an agent making any promise to receive information from the environment E, $a_i \xrightarrow{-\mathrm{env}_i} E$.

The study of real systems is therefore a study of leaky agents. The environment itself is also leaky in the sense that it can be affected by other agents. This is how we account for stigmergy for example.

With this view, we apply boundary conditions or coupling to the environment by giving every agent a use-promise from this environmental agent to allow some non-specified environmental conditions to be explicitly modelled. The environment agent is assumed to promise its information to all other agents. This is also the way to understand how agents making non-rigid promises can exhibit random behaviour. In order to justify random behaviour we must explain how disordered information enters the agents and selects values from within the bounds of the inexact promises. This is the only mechanism for exhibiting fluctuating behaviour.

By modelling forces using fictitious promises we can use the three laws above to explain all changes in a system in a common framework. Regardless of whether one finds this to one’s taste, it is a rather practical step for simple modelling. We add finally that the concept of a goal might now be extended to allow agents to desire outcomes about states in the environment, not only in their own state space. This is reasonable for any agent as long as it has a use-promise from another agent to accept changes of state. However, a goal is still not an elementary concept that can be the subject of a promise – it is an outcome that might emerge from the behaviour.

X. EMERGENT BEHAVIOUR AND GOALS

When is behaviour designed and when does it emerge? Promises are designed but outcomes emerge. Leaky agents especially can be influenced by environment and we cannot completely determine their trajectories. We speak of emergence when we identify behaviour that appears organized, but where no perceptible promises to account for this have been made.

Many authors have fallen into the trap of using the terminology of goals to describe emergence – goals which the parts of the system are incapable of knowing individually. This is a superfluous explanation which likely emerges from the fictitious belief that programming determines real world behaviour. We have shown that this is not the whole story and now offer a simple explanation for emergent organization.

To understand emergence we must look to the spectrum of observable outcomes of agents’ promises. Inexact promises allow for unpredictability and the question is to understand whether organized behaviour is likely, in spite of not being an agreed cooperative goal of the agents. We have proposed that promises must be inexact to allow for the possibility of unpredictable behaviour[26] and that the following simple

---

[4] In fact it is no more artificial than giving certain particles “charge” and defining the notion of a field in physics.

definition of emergent behaviour is plausible and captures the popular views in the literature.

Definition 7 (Emergent behaviour): Emergent behaviour is the set of trajectories belonging to leaky agents exhibiting non-rigid, collective behaviour that is observationally indistinguishable from organized behaviour.

The important issue here is observational indistinguishability. It is the end observer who looks for 'meaning' (i.e. a goal) in the organized outcomes; the actual promises made by the agents could in fact be anything that allows the observed outcome to arise. In other words an outcome 'emerges' simply because it arises.

There are many mysterious definitions of emergence in the literature but emergent behaviour can be understood easily by looking for any promises that enable the observed outcome and using algebraic reduction to account for behaviour as in section V-B, keeping firmly in mind the notion of observational indistinguishability. After all, if emergent behaviour is real, it should ultimately be measurable by the same standards as any other kind of behaviour.

The key to emergence then is that the residual freedoms in the agent promises (i.e. that are not constrained exactly), are selected from by interaction with the environment, resulting in patterns of behaviour that are unexpected, but which nevertheless lie within the bounds of the promises given.

An example of emergent behaviour often cited is the idea of a *swarm*. Many definitions of swarms have been offered[39], [40], [7], [41], [42], [43], [44], [45]. Ours is simply as follows:

Definition 8 (Emergent group or Swarm): A collection of leaky agents that may be seen by any external observer as exhibiting undifferentiated, collective behaviour.

XI. EMPIRICAL SUPPORT FOR PROMISES

There are ample studies to in the published literature in which to seek validation of at least some of the foregoing ideas. We propose to narrow our focus to just a few of these, since a full treatment would warrant a major study that is beyond the scope of the present paper.

The first part of our paper concerns laws of transmitted influence. The three laws themselves are proven from axiom and therefore the only validation required is of the assumptions on which they are based. Since the assumption is of autonomy, and one can always model a non-autonomous agent by an autonomous one armed with promises of submission, there is nothing worthy of verification. The laws are simply expressions of necessity, and we may turn the argument around to predict the existence of effective promises in all cases where influence is transmitted between components. Such promises are often represented as access control rules and network services. We encourage readers to be on the look-out for such promises in systems that seem to be under the control in a master-slave relationship. We are satisfied that they are present.

The importance of promises as opposed to events is their relative persistence, i.e. in allowing us to understand the stable properties of behaviour rather than passing transitions. A promise's outcome is represented by a distribution of outcomes rather than a single response to an event. Distributions have greater stability than individual observations. This is where the promise model differs from other works on the Event Condition Action Model. We expect to find predictability only at a statistical level, and have previously found this to be the case[46].

Apart from these minor verifications, we find most support from indirect studies of organization[47], [48], [49]. This is no doubt a result of contemporary predominance of interest in networks and their management. In this, two areas distinguish themselves for their clear dependence on *calibration* and *autonomous promises*: the Border Gateway Protocol and encryption key verification. Both of these subjects have been studied at length, especially the former. The data from these works are most useful in supporting ideas about organization and structural crystallization from peer-promises to centralized or hierarchical structures.

BGP studies are particularly interesting not for their routing at the level of packet events but because BGP behaviour has long term trends that are based on policies given by autonomous systems (AS). BGP policies are clearly promises by our definition concerning the transit of packets. Two phenomena are of interest: transit services and peering. Norton was amongst the first to consider the habits of BGP users[50], [51] and his result support the conclusion that the bindings made between AS's have little to do with packet traffic or transit tariffs, but rather everything to do with the potential value of their promises to peer with other powerful providers – in terms of social capital. Only when the cost of keeping these promises becomes debilitating do service providers waver from these promises. This is an explicit example of the importance of promises over events.

BGP also allows us to see delegated address spaces[49], which in promise terms implies a growth of autonomous agents and promises between them. The resulting structure of collaboration for sharing of the environmental resource is an organizational pattern. Sriraman et al. show that the structural organization of this is hierarchical from the top down. This kind of top-down phenomenon is not covered in our work because it occurs when a single agent splits into several agents with promises that link them to the residual of the original agent. Thus inevitably leads to a local cluster attached to an anchor point, but what is interesting is that the economics again drive the formation of a basically homogeneous tree in accordance with our predictions. The node degree of the graph is remarkably homogeneous, suggesting that a fixed number of promises is reached by a balance akin to that of eqn. (19).

Zhou et al.[48] make the point that this homogeneity is only a local phenomenon. The actual degree distribution of the BGP network follows a power-law behaviour with a long tail[34], [35]. Its average node degree is about 6, but maxima of up to 3000 are found. As we have pointed out, it makes sense for all but the richest resource providers to keep their promise counts low relative to their own capabilities – and these are not homogeneous. The most convincing support for our model comes from Norton's interpretation of the value to providers in bilateral peering[51]. He shows that the perceived value of outsourcing promises for major providers is a privately measured value and grows with increased peering relationships up to a maximum limit at which point it tails off, throttled by the resource bottleneck of the hub. This is mirrored in Dunbar's theory of human peering in anthropology[52].

A second area of computing in which organization structure is linked to economics of promises is in encryption key management. Here there are two basic models: direct key exchange, such as is used by tools like Secure Shell, and the trusted third party broker used by Transmission Layer Security SSL/TLS and Kerberos[31].

Dondeti et al. [53] provide some evidence to suggest that the cost distribution of key verification promises is already quite uniform, suggesting that the centralization bottleneck has been outpaced by improvements in technology. Clearly affordance can be sated either by a resource "arms race" or by diffusion of load.

The body of literature from economic organization theory is derived not only from economic game theory, but also from the observations made about organizations throughout its rather longer history, thus it brings a more complete and less artificial verification of promise predictions. A compelling survey of these ideas was found in the work of Fox [47], whose main points bear a striking resemblance to our results and therefore indirectly validate them: ours are based on simple axiomatic theory with few assumptions but predictive power, whereas his are based on experience of actual organizations. Human influence is prevalent in computer behaviour, since behind every computer system there lies a human decision-maker, vying for the value and success of the system.

If ref. [54] the authors define organizations as collaborative structures: "a group of persons who actions (decisions) agree with certain rules that further their common interests". Further they define teams: "an organization whose members have only common interests". We find these definitions typical of those in economic theory, and motivated more by wishful thinking than on the basis of an impartial model. To begin with, they are not founded on elementary concepts. Promise theory shows that cooperation is not an elementary concept but in fact requires a plethora of promises to accomplish. We have uncovered a deeper understanding based on simple arguments that both confirms prior experience and enhances it with mainly the assumption of autonomy at the heart.

There is much scope for future work in understanding the empirical verification of promise theory. Empiricism lies at its heart, so this is no small challenge. The magnitude of the task is also an indication of its actual substance.

XII. CONCLUSIONS AND FUTURE WORK

The term "behaviour" is wafted loosely in computer science with often little clarification. It is not about what we plan or desire, but about what actually happens when a system operates in its environment. Many questions about behaviour have been answered in the natural sciences. We have attempted to offer a usable description of the organization and behaviour based on the long scientific tradition of describing observable characters, within the language of promises. This has (unsurprisingly) many parallels with the physics of systems.

We have showed that behaviour is a pattern of change that can be partially predicted by reducing a system to an ensemble of autonomous agents making promises. Three basic laws of influence on behaviour follow from the property of autonomy that underlies promise theory. These laws are elementary expressions of change, explaining the promises required to transmit influence between autonomous agents; thus they describe the meaning of transmitted "force".

Observation is the cornerstone to understanding agent behaviour because promises never imply guaranteed determinism, only distributions of outcomes that can be observed in experimental trials. Our ability to distinguish agents stems from our ability to distinguish their behaviour, either in advance (from promises) or after the fact (from their observed trajectories). The ability to observe differences in behaviour is not guaranteed: there are symmetries in ensembles of indistinguishable agents that require the calibration scale of a single adjudicating observer to gauge. Once a scale of distinctions can be made, the concept of organization can be explained empirically, purely in terms of observable patterns of variation, without the need to imagine that they are always the result of complex human concepts like 'designs' or 'goals'. The 'self-organization' is a redundant terminology, as organization is a measurable property of any system.

A frequently emerging pattern is the hierarchy. We argue that this does not emerge out of the need for control or separation of concerns as do other authors, but rather from the avoidance of economic cost associated with observing and distinguishing system components on a single calibrated scale of measurement: the comparison of capabilities. We cite examples from BGP and key-signing in support of this.

We advise readers, having read this paper, to quell the urge to think of promises as a network protocol, or even as message passing. The promise graph is not a map of the network but an abstract set of relationships whose message passing medium is not necessarily known. Structural or organizational relationships do not have to occur through regular interaction as long as the agents can remember their promises. Once established, promises persist like intrinsic properties.

The descriptions we offer here are a platform from which to clarify many issues in autonomic systems. There are unanswered questions about the subjective nature of agent perceptions that motivate the need for a full theory of measurement based on promise agents. This work is only the beginning. From such a theory it should be possible to decide whether peer to peer and centralized systems are comparable organizations with interchangeable properties, or whether they are two fundamentally different things. This is not clear today.

We believe that it is possible to go further and define material properties for promise graphs, by analogy to how physics describes the large scale properties of matter from an atomic model. Why is wood strong and glass brittle? Why is one computational structure robust and another fragile? These are analogous questions that are about scale as well as the underlying promises that bind the parts into a whole. We must work towards suitable and useful definitions of these properties. We believe that this such definitions must follow from promise theory or something like it. We return to these issues in future work.

MB is grateful to Jan Bergstra, Matt Disney and Samah Bel Haj Saad for helpful discussions. This work is supported by the EC IST-EMANICS Network of Excellence (#26854)

REFERENCES

[1] M. Burgess and S. Fagernes. Laws of systemic organization and collective behaviour in ensembles. In *Proceedings of MACE 2007*, volume 6 of *Multicon Lecture Notes*. Multicon Verlag, 2007.

[2] M. Burgess. On the theory of system administration. *Science of Computer Programming*, 49:1, 2003.

[3] M. Burgess. Configurable immunity model of evolving configuration management. *Science of Computer Programming*, 51:197, 2004.

[4] A. Couch and Y. Sun. On the algebraic structure of convergence. *LNCS, Proc. 14th IFIP/IEEE International Workshop on Distributed Systems: Operations and Management, Heidelberg, Germany*, pages 28–40, 2003.

[5] A.L. Couch and M. Chiarini. Dynamic consistency analysis for convergent operators. In *Lecture Notes on Computer Science: Resilient Networks and Services*, volume 5127, pages 148–161, 2008.

[6] A.L. Couch and M. Chiarini. A theory of closure operators. In *Lecture Notes on Computer Science: Resilient Networks and Services*, volume 5127, pages 162–174, 2008.

[7] M. Wooldridge. *An Introduction to MultiAgent Systems*. Wiley, Chichester, 2002.

[8] J.L. Hellerstein, Y. Diao, S. Parekh, and D.M. Tilbury. *Feedback Control of Computing Systems*. IEEE Press/Wiley Interscience, 2004.

[9] N. Damiannnou, A.K. Bandara, M. Sloman, and E.C. Lupu. *Handbook of Network and System Administration*, chapter A Survey of Policy Specification Approaches. Elsevier, 2007 (to appear).

[10] Mark Burgess. An approach to understanding policy based on autonomy and voluntary cooperation. In *IFIP/IEEE 16th international workshop on distributed systems operations and management (DSOM), in LNCS 3775*, pages 97–108, 2005.

[11] R. Coase. The nature of the firm. *Economica*, 4(16):386–405, 1937.

[12] T. Peters and R.H. Waterman Jr. *In Search of Excellence*. Profile Books, 1982,2003.

[13] J.L.G. Dietz. *Enterprise ontology*. Springer, 2006.

[14] Feng Wan and Munindar P. Singh. Commitments and causality for multiagent design. In *Proceedings of the 2nd International Joint Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*, 2003.

[15] Feng Wan and Munindar P. Singh. Formalizing and achieving multiparty agreements via commitments. In *International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2005.

[16] J. Bergstra and I. Bethke. A process algebra based framework for promise theory. Technical report, University of Amsterdam, 2007.

[17] Munindar P. Singh. Semantical considerations on dialectical and practical commitments. In *Proceedings of the 23rd Conference on Artificial Intelligence (AAAI).*, 2008.

[18] G.D. Rodosek, H.G. Hegering, and B. Stiller. Dynamic virtual organizations as enablers for managed invisible grids. In *Proceedings of the IEEE/IFIP Network Operations and Management Symposium (NOMS)*, 2006.

[19] I. Fosterand C. Kesselman and S. Tuecke. The anatomy of the grid: Enabling scalable virtual organizations. *International Journal of Supercomputer Application*, 15(3):200 222, 2001.

[20] M. Burgess and S. Fagernes. Pervasive computing management: A model of network policy with local autonomy. *IEEE Transactions on Software Engineering*, page (submitted).

[21] M. Burgess and S. Fagernes. Voluntary economic cooperation in policy based management. *IEEE Transactions on Network and Service Management*, page (submitted).

[22] E. Lupu and M. Sloman. Conflict analysis for management policies. In *Proceedings of the Vth International Symposium on Integrated Network Management IM'97*, pages 1–14. Chapman & Hall, May 1997.

[23] M. Burgess. *Analytical Network and System Administration — Managing Human-Computer Systems*. J. Wiley & Sons, Chichester, 2004.

[24] J.M. Hendrickx et al. Rigidity and persistence of three and higher dimensional forms. In *Proceedings of the MARS 2005 Workshop on Multi-Agent Robotic Systems*, page 39, 2005.

[25] J.M. Hendrickx et al. Structural persistence of three dimensional autonomous formations. In *Proceedings of the MARS 2005 Workshop on Multi-Agent Robotic Systems*, page 47, 2005.

[26] M. Burgess and S. Fagernes. Norms and swarms. *Lecture Notes on Computer Science*, 4543 (Proceedings of the first International Conference on Autonomous Infrastructure and Security (AIMS)):107–118, 2007.

[27] M. Burgess and A. Couch. Autonomic computing approximated by fixed point promises. *Proceedings of the 1st IEEE International Workshop on Modelling Autonomic Communications Environments (MACE); Multicon verlag 2006. ISBN 3-930736-05-5*, pages 197–222, 2006.

[28] H. Lewis and C. Papadimitriou. *Elements of the Theory of Computation, Second edition*. Prentice Hall, New York, 1997.

[29] R. Axelrod. *The Complexity of Cooperation: Agent-based Models of Competition and Collaboration*. Princeton Studies in Complexity, Princeton, 1997.

[30] R. Axelrod. *The Evolution of Co-operation*. Penguin Books, 1990 (1984).

[31] M. Bishop. *Computer Security: Art and Science*. Addison Wesley, New York, 2002.

[32] R. Albert and A. Barabási. Statistical mechanics of complex networks. *Reviews of Modern Physics*, 74:47, 2002.

[33] D.J. Watts. *Small Worlds*. (Princeton University Press, Princeton), 1999.

[34] M. E. J. Newman, S.H. Strogatz, and D.J. Watts. Random graphs with arbitrary degree distributions and their applications. *Physical Review E*, 64:026118, 2001.

[35] A.L. Barabási. *Linked*. (Perseus, Cambridge, Massachusetts), 2002.

[36] G. Canright and K. Engø-Monsen. A natural definition of clusters and roles in undirected graphs. *Science of Computer Programming*, 53:195, 2004.

[37] W.D. McComb. *Renormalization Methods: A Guide for Beginners*. Oxford University Press, 2003.

[38] A.L. Barabási and R. Albert. Emergence of scaling in random networks. *Science*, 286:509, 1999.

[39] E. Bonabeau, M. Dorigo, and G. Theraulaz. *Swarm Intelligence: From Natural to Artificial Systems*. Oxford University Press, Oxford, 1999.

[40] J. Kennedy and R.C. Eberhart. *Swarm Intelligence*. Morgan Kaufmann (Academic Press), 2001.

[41] G. Di Caro and M. Dorigo. Antnet: Distributed stigmergetic control for communications networks. *Journal of Artificial Intelligence Research*, 9:317–365, 1998.

[42] L. Arlotti, A. Deutsch, and M. Lachowicz. On a discrete boltzmann type model of swarming. *Math. Comp. Model*, 41:1193–1201, 2005.

[43] Kazadi S. *Swarm Engineering*. PhD thesis, California Institute of Technology, 2000.

[44] F. Heylighen. *Open Source Jahrbuch*, chapter Why is Open Access Development so Successful? Stigmergic organization and the economics of information. Lehrmanns Media, 2007.

[45] J.H. Holland. *Emergence: from chaos to order*. Oxford University Press, 1998.

[46] M. Burgess. Evaluation of cfengine's immunity model of system maintenance. *Proceedings of the 2nd international system administration and networking conference (SANE2000)*, 2000.

[47] M.S. Fox. An organizational view of distributed systems. *IEEE Transactions on Systems, Man and Cybernetics*, SMC-1:70–80, 1981.

[48] S. Zhou and R.J. Mondragon. Analyzing and modelling the AS-level Internet topology. *ArXiv Computer Science e-prints*, March 2003.

[49] A. Sriraman, K.R.B. Butler, P.D. McDaniel, and P. Raghavan. Analysis of the ipv4 address space delegation structure. *Computers and Communications, 2007. ISCC 2007. 12th IEEE Symposium on*, pages 501–508, July 2007.

[50] W.B. Norton. The art of peering: The peering playbook. Technical report, Equinix.com, 2001.

[51] W.B. Norton. Internet service providers and peering. Technical report, Equinix.com, 2001.

[52] R. Dunbar. *Grooming, Gossip and the Evolution of Language*. Faber and Faber, London, 1996.

[53] L. Dondeti, S. Mukherjee, and A. Samal. Survey and comparison of secure group communication protocols, 1999.

[54] J. Marschak and R. Radner. *Economic Theory of Teams*. Yale University Press, 1972.