Programming Research Group

TWO PAPERS ON THE FOUNDATIONS OF COMPUTER ETHICS:

ENTROPY AS EVIL IN INFORMATION ETHICS

ARTIFICIAL EVIL AND THE FOUNDATIONS OF COMPUTER ETHICS

Luciano Floridi and J.W.Sanders

PRG-TR-05-00

Oxford University Computing Laboratory
Wolfson Building, Parks Road, Oxford OX1 3QD

Entropy as Evil in Information Ethics

Luciano Floridi and J. W. Sanders

Abstract

It has been argued [5, 6] that Information Ethics, which amongst other things concerns questions of normally non-sentient agents acting on information (i.e. entities in Cyberspace), is a *macroethics* (a theoretical, field-independent, applicable ethics) not accurately subsumed by standard ethical theories. In this paper that argument is extended. Four requirements of Information Ethics are determined: stability, modularity, rigorousness and soundness. A concept called *entropy structure* is proposed as the basis for an Information Ethics consistent with those requirements. Finally, evil actions are characterised as those that increase the entropy ordering and a variety of examples is analysed.

From *Etica & Politica*, Monographica "Computer Ethics", December 1999.

1 Introduction

Imagine a sufficiently simple world in which each being reacts to stimulii and performs actions by (a) observing certain values around it and (b) updating its internal state according to some deterministic rule. Perhaps the internal states of other beings are discernible and so they too provide values to be employed in that process.

In principle we could predict each agent's behaviour, if we knew each rule, by simulating the process described above. All past behaviour would be encapsulated in an agent's internal state. All future behaviour would be determined from an agent's current state by the parameters to which it becomes subjected. Thus all consequences of an action could in pricinple be determined. Since internal state is revealed, there would be little use in that world for terms we use in our world to approximate the result of our own mental state (like mood, obligation, regret, intent and so on). Indeed the study of Ethics in such a world would be radically different from that in ours.

Is it conceivable that our world is like that? Many of the corresponding values involved are subjective. They result from agents whose internal states are far from clear even to themselves, let alone to others. And so it might be felt that such a world, no matter how interesting its lessons might be for our own Ethics, could only be the object of a remote Gedanken experiment (c.f. [15]).

In this paper we begin an investigation of the degree to which Cyberspace — that world of information acted upon by artificial agents — can be regarded as such a world. There are convincing reasons why it might be. Each agent can be thought of as a finite automaton which acts in just the way described above: its state is determined by the values of all its internal variables; it reacts with its environment by consuming input and generating output; and its program code determines the rule by which it updates its state in light of its input and by which it acts. On the other hand, there are bound to be limitations on the degree to which that view tells the whole story about Cyberspace. If a formula does exist to answer yes/no questions about the ethical nature of an action in Cyberspace, it could not be computable; for the uncomputable Halting Problem can be encoded as such a question. One could not expect to answer low-level engineering or physics questions about the media in which the agents are implemented because the nondeterminism involved in quantum phenomena would defeat any determinismtic formula. And care would have to be exercised to circumscribe the boundary of Cyberspace, otherwise subjective values of its human users would easily supervene (e.g. 'I prefer one program to another because its interface suits me better').

With these provisos in mind, our interest in such an investigation arises from the pressing need for a methodology to discuss problems of Computer Ethics (the supply of examples of which is enormous; see [1, 13]). In this paper, we concentrate on problems restricted to Cyberspace, overlooking human intervention. In that domain, we develop the suggestion [5] that entropy provides a foundation for Information Ethics — the methodological foundation of Computer Ethics. The larger context of our work is the widespread interest in artificial equivalents, in

Cyberspace, of standard concepts like epistemology (Pollock), metaphysics (Steinhart), life (Bedau, Boden), experience (Lycan), morality (Danielson), creativity (Boden) and of course intelligence and learning (see [2, 11] for a current summary of those endeavours). We shall find ourselves promoting a new interpretation of a form of evil distinct from moral or natural evil. We label it 'artificial'. Artificial evil appears suitable for modelling immoral or criminal actions in Cyberspace. Our task here is to consider the foundation of an Information Ethics that may deal with artificial evil; the subject is studied in its own right in [7].

We begin by recalling, in section 2, why Information Ethics is not a standard Ethics. In section 3 we determine the properties an Information Ethics should have; and in section 4 we introduce methods for defining entropy structures in Cyberspace. An action is regarded as a state transformer and thereby judged evil if it increases state in the entropy ordering. Although our methodology is formal (the action is described mathematically and its propensity for evil decided by a mathematical definition) ordinary judgement underlies the choice of ordering. Naturally, the assumptions which make that approach viable in Cyberspace do not necessarily hold in the 'real world'. Examples from both Cyberspace and the real world are considered in section 5. In section 6 we reflect on why our proposal is consistent with the properties we formulated earlier.

We begin by discussing the untenability of standard Ethics in any formulation of Information Ethics.

2 Information Ethics is a non-standard Ethics

Agents in Cyberspace perform actions with far-reaching consequences. Increasingly, they are embedded in larger systems which invoke them autonomously. Think of control of safety-critical systems (like fly-by-wire aircraft, train routing, power plants and air-traffic control), management of sensitive or even secret data (like the electoral register and commercial or government databases responsible for national security), communications systems (like email, file transfer and ecommerce) and expert systems (medical, scientific or legal).

Because they no longer need to be the result of a single programming project, and they may not be directly under control of a human, the morality of the actions of such agents seems to require formulation and analysis. For example, was the action of the autopilot wrong? Who is to blame? Is the encryption software good? Ought an agent to be able to eavesdrop on this email conversation? Does my word processor behave correctly with respect to its autonomous duties concering fonts, pagination, footnotes, spelling, cross referencing, and so on? Even a simple, non-moral question such as "did *Deep Blue* play well?" is loaded with evaluative assumptions.

A domain of discourse and set of values is required for the discussion of such questions. We are not here concerned so much with how that is used. We imagine it being employed much as in standard Ethics, to reason in levels of detail. ('Life is precious' is a standard ethical rule at a low level of detail; a particular murder might be judged moral by appeal to some rule at a finer level, like 'liberation of oppressed populations is good'. We shall see that entropy, in its most general form, provides a way of reasoning that viral action in Cyberspace is in general bad because it destroys structure; but entropy in a finer form might be used to reason for the benevolence of a virus which appears equally random but on closer analysis turns out to perpetrate patches to fix millenial bugs.)

An initial attempt has been, naturally, to ingore Cyberspace (because of its novelty?): anthropic principles have been used either to shift the burden of an agent's actions to its creators, or to treat it as if it were in some way sentient. Whilst for simple programs on a stand-alone machine that view may be partially acceptable, for those whose code accrues over the web, whose behaviour is modified over time (so called 'learning' programs [11]) and use probabilistic choices [12] it is simply not feasible (for a constantly updated supply of topical examples which demonstrate the inadquqcy of standard macroethics in reasoning about evil in Cyberspace see [13] and the articles in this issue of *Etica & Politica*. Perhaps that is why some computer scientists have, for many years, explicitly resisted anthropism, claiming it to be the sign of an immature discipline [4]!

Floridi [6] has suggested that information be elevated in status to 'being' and [5] that entropy be used to discuss moral claims concerning action on it. He argues that standard Ethics is unable, by itself, to provide an appropriate foundation for Computer Ethics. In the specific case represented by the non-sentient constitution of Cyberspace, this renders the laws of standard (inevitably anthropocentric) Ethics inapplicable and its conclusions remote whilst providing a vehicle for crime which may be victimless or of a characteristically ludic nature. Standard or classic macroethical theories like Virtue Ethics, Consequentialism and Deontologism, being anthropocentric and agent-oriented, take only a relative interest in the 'patient', which is on the receiving end of the action and endures its effects. Only non-standard approaches, like Medical Ethics, Bioethics and Environmental Ethics, attempt to develop full patient-oriented ethics in which the 'patient' may be any form of life. They argue that the nature and well-being of the patient of an action constitute its moral standing and that the latter makes vital claims on the interacting agent and ought to contribute to the guidance of the agent's ethical decisions and the constraint of the agent's moral behaviour. Compared with standard Ethics, non-standard Ethics are theories of nature and space — their ethical analyses start from the moral properties and values of what there is — no longer of history and time (human actions and their consequences). By placing the 'receiver' of the action at the centre of the ethical discourse, and displacing its 'transmitter' to its periphery, they help to widen further our anthropocentric view of who or what may qualify as a centre of moral concern. Classic Ethics is inevitably egocentric and logo-centric: all theorising concerns a conscious and self-assessing agent whose behaviour must be supposed sufficiently free, reasonable and informed for an ethical evaluation to be possible on the basis of his or her responsibility. By contrast non-classic ethics, being bio-centric and patient-oriented, are epistemologically allocentric: they are centred on, and interested in, the entity itself that receives the action, rather than in its relation or relevance to the agent

— and morally altruistic, so they can include any form of life and all vulnerable human beings within the ethical sphere (not just foetuses, new-born babies and senile persons, but above all physically or mentally ill, disabled or disadvantaged people).

From this perspective, it can be argued that Information Ethics is the last stage in the development of a non-standard approach, an ontocentric expansion of environmental ethics towards a fully non-biologically biased concept of a 'centre of moral worth' [6].

3 Requirements of Information Ethics

Floridi [5] has identified general properties characteristic of Computer Ethics, finding it to be based on case studies, to be intrinsically decision-making oriented, to endorse a problem-solving approach, to be empirically grounded and to be logically argumentative. In this section we investigate what constraint those and other properties place on Information Ethics, the methodological foundation of Computer Ethics. We shall find that Information Ethics should satisfy four properties we call stability, modularity, rigorousness and soundness.

3.1 Stability

The rapid evolution of Information Technology has meant the equally rapid expansion of Cyberspace. Tasks previously performed entirely by humans are now partially or fully computer controlled. Examples include expert systems (e.g. autopilots, medical systems and game-playing programs like *Deep Blue*), workplace software (e.g. spreadsheets, editors with spelling checkers, databases and graphics packages) and other applications software (e.g. email, search engines, hypertext, webbots and ecommerce). Important advances in Information Technology with direct impact on users have included the extension from isolated microprocessors to networks spanned by the web, the extension from batch processing to autonomous agents which even learn by modifying their behaviour in the light of their own 'experience', and the advance in interface design from punch cards to palm devices which adapt to the user's handwriting and to vocally-operated operating systems.

A foundation for Computer Ethics ought not to require alteration with changes in Information Technology. For example, had it been developed thirty years ago, Information Ethics would have needed to have been robust against the advances mentioned in the previous paragraph. Information Ethics must be *stable*.

3.2 Modularity

Information Systems, the components of Cyberspace, are the most complex engineering products yet produced. Their components range from physical devices, like transistors, which require careful design and manufacture through operating systems and applications packages to the intricacies of the internet and world-wide web. To tame that complexity *modularisation* is required: a system is designed in modules each responsible for only part of the overall functionality and therefore able to be produced largely independently of the rest of the system. The coarsest division in the hierarchy of modules follows the very terms 'hardware' and 'software'. At the programming level, a module is a procedure invoked by a higher structure.

Many (some would argue *all*) modules could, in principle, be performed instead by a human with consequent change only in efficiency, not functionality (for example spelling checking, piloting an aircraft or otherwise behaving like an expert, performing arithmetic and balancing a spreadsheet). Indeed historically, advances in Information Technology might be viewed as the automation of routine human behaviour (or even in some cases, like chess, not considered routine). A typical example of such modular replacement occurs in Searle's Chinese Room [14] in which human comprehension — the standard factor at stake there — is argued not to follow from manual execution of an otherwise-automated module.

To reflect the way in which information systems are constructed, understood and executed, Information Ethics must support modular and incremental reasoning. If one module is performed differently, the analysis of the whole system should be modified by replacing only the argument concerning that single module, not begun again from scratch. For example if manual operation of a module moves it outside Cyberspace, the corresponding piece of argument should be removed and the remainder reworked without it. Information Ethics must be *modular*.

3.3 Rigorousness

The complexity of Cyberspace results from the large number of states possessed by its components, which account for the multifarious and often subtle ways in which those components act and interact. That is why programs are treated as mathematical objects [8] and why Formal Methods plays such an important rôle in the development of critical systems. Their complexity means that even software testing is unable to establish the correctness of any but the most trivial systems. Attempts to reason informally about the correctness of algorithms are well known to fail spectacularly. An ethical theory based on informal reasoning would be unable to account for the multitude of cases which arise from the complexity of a system's state space.

We conclude that to account for the decision-making, problem-solving and logically argumentative nature of Computer Ethics, a methodological foundation is required which supports rigorous reasoning. Of course it should, like any formal methodology, be *logically consistent* to escape vacuous conclusions of logical implications.

3.4 Soundess

Since Computer Ethics is empirically grounded, Information Ethics must lay the foundation for a sensible analysis of typical case studies: it must be *sound*. The most obvious of our conditions, soundness excludes trivial methodologies, like that in which each statement is deemed to be true (which would satisfy the remaining three conditions). It needs to be stated, therefore, only because of the rigorous nature of the methodology.

The seemingly simple condition of soundess has shaped the approach we take. For it implies that there must be a place in Information Ethics for codification of certain of our values: those we employ when calibrating the sense of an example. One person's junk email is another's treasured correspondence; so any foundation for Computer Ethics must be able to draw that distinction. No absolute interpretation can alone be sufficient, any more than it is in standard Ethics. Levels of reasoning must be possible, with rules of finer detail overriding more general ones.

That important point is reinforced by the following example. Suppose we design a virus to attack all computers on the net. It has the property that at say the dawn of 2000 the picture of Mona Lisa appears, in some encoded form, distributed throught the world's filestores. Your computer may contain part of her left eyebrow but in a form unrecognisable to you, simply encoded in 0's and 1's. Locally, at each processor, my virus has increased entropy. But globally it has imposed structure where before there was none, and so has decreased entropy. Were that technique used to request help from extraterrestial beings in the event of global disaster it would no doubt be regarded as beneficial; but as a ludic exercise it is evil. Ordinary judgement must be taken into account before evil actions can be formalised in Cyberspace. Information Ethics shares this fundamental requirement with all other possible ethical approaches.

4 Entropy-based Information Ethics

Entropy has been introduced in [5] as a vehicle for expressing four laws that govern the minimal level of morality of an action:

0. entropy ought not to be caused in the infosphere (null law)

1. entropy ought to be prevented in the infosphere

2. entropy ought to be removed from the infosphere

3. information welfare ought to be promoted by extending (information quantity), improving (information quality) and enriching (information variety) the infosphere.

Here we limit our interest to Cyberspace (not the whole infosphere) and investigate possible formalisations of that idea and how it may be used in judging actions to be evil.

In thermodynamics and communication theory entropy has an absolute definition reflecting the *variety* of possible states. However in order to use that idea in Cyberspace *modularity* and *soundness* require us to provide a definition that is flexible enough to describe an action as entropy-increasing in one situation but not another. Someone regarded as a valid recipient of email in one situation is regarded in another as an eavesdropper. A change to our filestore is regarded as beneficial in one situation but viral in another. Although the contents of Cyberspace are themselves entirely objective, our evaluation of them is naturally subjective. Indeed without an *a priori* view of what constitues good and bad structure in Cyberspace, the notion of entropy and entropy-increasing action is in danger of becoming circular. Our proposal is to capture such a view mathematically in a concept called an *entropy structure* with respect to which *evil* actions are entropy increasing and *good* actions entropy decreasing.

4.1 Entropy structures

It is the approach and practice of *Formal Methods* to describe any part of Cyberspace mathematically (popular model-based formalisms consistent with our treatment here are given by Z [16] and VDM [10]). Here we adopt that approach to conceptualise the part of Cyberspace, in which we are interested at a particular time, as an *object* (or *module*, or *abstract data type*) having a set of precisely-defined *states* operated on by a family of mathematically specified *actions* (or *methods* or *operations*). Examples are provided in section 5.

Let X denote the set of states — the state space — of the system under consideration. It could be one small component, like an editor on a specific machine with the actions it offers the user, or a network with many terminals and filestores connected to the internet and all its attendant actions. In the extreme case it is the whole of Cyberspace. We now introduce the notion of *entropy structure* on X. The elements in the definition are routine (see, for example, [3]).

We use the following notation. The symbol $=_{def}$ means 'equals by definition'. All variables of each predicate are assigned a type and quantification is over the appropriate type. The predicate

$$\forall x : X \bullet P$$

is read *for all* x *in* X*,* P *holds*. Thus the symbol $\bullet$ is used to separate the typed quantifiers from the body of the predicate. We write

$$P \Rightarrow Q$$

for the predicate P *implies* Q and

$$X \longrightarrow Y$$

for the space of all functions from X to Y. Otherwise our notation is standard.

A relation A on X is a subset of the Cartesian product of X with itself

$$A \subseteq X \times X.$$

Membership of an ordered pair to relation A

$$(x, x') \in A$$

is read A *relates* x *to* x' and written in *infix*:

$$x\,A\,x'.$$

The *converse* $A^{\sim}$ of a relation A on X is its 'mirror image'

$$x\,A^{\sim}\,x' \;=_{def}\; x'\,A\,x.$$

A *pre-order* on X consists of a relation $\leq$ on X which is reflexive (i.e. includes the identity relation) and transitive (i.e. closed under sequential composition):

$$\forall\, x : X \bullet x \leq x$$
$$\forall\, x, y, z : X \bullet (x \leq y) \wedge (y \leq z) \Rightarrow x \leq z.$$

A pre-order $\leq$ is *total* iff any two elements of X are related by either $\leq$ or its converse

$$\forall\, x, x' : X \bullet x \leq x' \;\vee\; x' \leq x.$$

The equivalence relation $\sim$ of a pre-order $\leq$ is the relation on X equal to the intersection of $\leq$ and its converse

$$x \sim x' \;=_{def}\; (x \leq x') \wedge (x' \leq x).$$

To be an equivalence relation means that it is not only a pre-order on X, but also *symmetric* (i.e. contained in its converse):

$$\forall\, x, x' : X \bullet x \sim x' \Rightarrow x' \sim x.$$

The $\sim$-*equivalence class* of $x : X$ consists of all elements of X equivalent to x

$$x^{\sim} \;=_{def}\; \{y : X \mid x \sim y\}.$$

The definition of pre-order is sufficient to ensure that the equivalence classes of elements of X *partition* X:

$$\cup\{x^{\sim} \mid x \in X\} = X$$
$$x^{\sim} \cap y^{\sim} \neq \{\,\} \quad\Rightarrow\quad x^{\sim} = y^{\sim}.$$

Furthermore the pre-order $\leq$ is well-defined on equivalence classes: the relation, again called $\leq$, which is defined on equivalence classes if the original pre-order holds between representative elements of those classes

$$x^{\sim} \leq y^{\sim} \;=_{def}\; x \leq y$$

is well-defined (i.e. does not depend on the representatives), is again a pre-order, and in addition is *antisymmetric*:

$$\forall\, x, y : X \bullet x \leq y \wedge y \leq x \Rightarrow x \sim y.$$

In other words $\leq$ becomes a *partial order* on $\sim$-equivalence classes.

An important way to define a pre-order $\leq$ on X is to define a *level function* λ from X to the real numbers, $\lambda : X \longrightarrow \mathbb{R}$, and then to define the derived pre-order $\leq_\lambda$ by setting

$$x \leq_\lambda x' \ =_{def} \ \lambda(x) \leq_{\mathbb{R}} \lambda(x') \tag{1}$$

where the order $\leq_{\mathbb{R}}$ denotes the standard (total) ordering between real numbers. Pre-orders so defined are automatically total partial orders, by properties of $\leq_{\mathbb{R}}$. In fact because Cyberspace is discrete by choice of an appropriate encoding we can always ensure that $\mathbb{R}$ there is replaced by the natural numbers $\mathbb{N}$ whose ordering $\leq_{\mathbb{N}}$ is a restriction of $\leq_{\mathbb{R}}$. But use of $\mathbb{R}$ permits us the flexibility of considering also continuous examples from the 'real' world.

By an *entropy structure* we mean a triple $(X, \leq, \sim)$ consisting of a set X, a pre-order $\leq$ on X and the equivalence relation $\sim$ of $\leq$.

It is worth emphasising that the definitions of the ingredients of an entropy structure can be couched in whatever mathematical notions are convenient. Typically level functions and pre-orders are expressed using concepts perceived only from 'outside' the system under consideration. That ought not to be surprising: whilst the subsystem of Cyberspace must be self-contained its description is a meta-activity.

4.2 Entropy-changing actions

A terminating action on state space X takes a state-before x to a state-after x' and so is modelled as a relation on X. If the action is deterministic then for each state-before there is only one state-after and so the relation is a function.

Let $(X, \leq, \sim)$ be an entropy structure. An action A on X is *entropy decreasing* iff A is contained in the converse of $\leq$

$$\forall\, x, x' : X \bullet x\, A\, x' \Rightarrow x' \leq x.$$

In other words whenever the action of A can change the state of the system from an initial state to a final state, the final state must be smaller (in the given order). We refer to such an action more briefly and informally as *good*.

Action A is *entropy increasing* iff it intersects $\leq$

$$\exists\, x, x' : X \bullet x\, A\, x' \wedge x \leq x'.$$

In other words it is possible to start the action A in some state such that A leaves the system in a larger state (in the given order). We refer to such an action informally as *evil*.

Finally, A leaves entropy *invariant* iff it is contained in $\sim$

$$\forall\, x, x' : X \bullet x\, A\, x' \Rightarrow x \sim x'.$$

In other words action of A always leaves the system in an equivalent state (from the viewpoint of the given order). Such an action we refer to as *benign*.

The asymmetry between the definitions of entropy increasing and entropy decreasing reflects the intuition that one 'bad' instance is enough for an action to be judged 'evil'; but that to be judged 'good' all of its instances must be 'good'.

For two actions A_0 and A_1 on X we say that A_1 is *more evil* than A_0 iff from any given state A_1 results in a state with higher entropy than does A_0

$$\forall\, x, x', x'' : X \bullet (x\, A_0\, x' \wedge x\, A_1\, x'') \;\; \Rightarrow \;\; x' \leq x''.$$

4.3 Combining entropy structures

Those definitions apply, as we have said, to any subsystem of Cyberspace. We now show how they may be combined and so used to define complicated systems in terms of simpler ones. There are two requirements. We must be able to combine simple views of a given system to endow it with a more realistic complex view; and we must be able to combine different systems to describe a larger one equally exactly.

If $\leq_0$ and $\leq_1$ are pre-orders on X then their *conjunction* is defined to be their intersection

$$x \leq_\wedge x' \;\; =_{def} \;\; x \leq_0 x' \wedge x \leq_1 x'.$$

Conjunction corresponds to combining the two orders independently but equally on a given state space. It is clearly again a pre-order whose equivalence is the intersection of the two equivalences

$$x \sim_\wedge x' = x \sim_0 x' \;\; \wedge \;\; x \sim_1 x'.$$

Thus we define the *conjunction* of two entropy structures $(X, \leq_0, \sim_0)$ and $(X, \leq_1, \sim_1)$ on the same state space to be the entropy structure $(X, \leq_\wedge, \sim_\wedge)$.

The *lexical combination* in which pre-order $\leq_0$ is refined by pre-order $\leq_1$ on the same state space, employs (as the name suggests) the latter to refine the former

$$x \leq_{lex} x' \;\; =_{def} \;\; (x \leq_0 x') \vee (x \sim_0 x' \wedge x \leq_1 x')$$

where $\sim_0$ denotes the equivalence of $\leq_0$. Investigation of cases shows it to be again a pre-order. Immediately from the definition its equivalence is seen to be

$$x \sim_{lex} x' = x \sim_0 x' \wedge (x \leq_1 x' \;\; \vee x' \;\; \leq_1 x).$$

Thus we define the *lexical combination* of two entropy structures $(X, \leq_0, \sim_0)$ and $(X, \leq_1, \sim_1)$ on the same state space to be the entropy structure $(X, \leq_{lex}, \sim_{lex})$.

For pre-orders $(X_0, \leq_0)$ and $(X_1, \leq_1)$ on possibly different state spaces their product is the pre-order $\leq_\times$, on the Cartesian product $X_0 \times X_1$ of the state spaces, defined

$$(x_0, x_1) \leq_\times (y_0, y_1) \;\; =_{def} \;\; x_0 \leq_0 y_0 \wedge x_1 \leq_1 y_1.$$

It is clearly again a pre-order whose equivalence is, like that for conjunction,

$$(x_0, x_1) \sim_\times (y_0, y_1) = x_0 \sim_0 y_0 \wedge x_1 \sim_1 y_1 .$$

That leads us to define the *product* of two entropy structures $(X_0, \leq_0, \sim_0)$ and $(X_1, \leq_1, \sim_1)$ to be $(X, \leq_\times, \sim_\times)$.

For examples of the way in which a complicated ordering can be obtained on a state space by combining several simple orders using those combinators, see section 5.4.

5 Examples

We now analyse five examples in terms of entropy structure. The first four are chosen from primary components of Cyberspace: a communications medium, a virus, a webbot and an editor. The first three of those demonstrate the classification of good, evil and benign actions; the fourth shows how more complex entropy structures may be defined in terms of simple ones. The fifth example is chosen to show, for comparison, how entropy structures fare in analysing situations outside Cyberspace, in the real world.

5.1 Communication

We consider a subsystem of Cyberspace that models two computers in communication. Data are transmitted from sender to receiver through an intervening medium M by communication action C. Action C may result in corrupt data being received, in practice because medium M is imperfect. Let us see how an entropy structure may be defined in order to quantify the amount of corruption achieved — i.e. evil perpetrated — by C.

Let D denote the set of data under consideration and let D^* denote the set of all finite sequences of elements of D. The state of the system consists of a pair (s, r) of sequences of data, having the same length. Sequence s represents the data transmitted so far and sequence r represents the data received so far. Both sequences accumulate as time evolves and reflect the order in which data are handled. They have the same length because we choose to describe the system only after a datum has been received but before the next has been transmitted (since we here have no interest in observing the state of the system whilst data are in transit). Thus the set of system states is

$$X \ =_{def} \ \{(s, r) : D^* \times D^* \mid \#s = \#r\}$$

where $\#t$ denotes the size (i.e. length) of finite sequence t.

The communication action C is represented relationally as follows. If datum d is communicated successfully then

$$(s, r)\ C\ (s.d, r.d)$$

where $t.d$ is the sequence obtained by placing item d at the end of sequence t. However if datum d is corrupted to d' then

$$(s, r)\ C\ (s.d, r.d').$$

The pre-order we choose on X is defined using the level function $\lambda : X \longrightarrow \mathbb{N}$ which is itself defined to reflect the number of messages corrupted

$$\lambda(s, r) \ =_{def} \ \#\{j \ \mid \ r_j \neq s_j\}$$

where r_j and s_j denote the jth elements of sequences r and s respectively and $\#S$ denotes the size (i.e. cardinality) of finite set S.

With respect to the pre-order defined by equation (1) we have now defined an entropy structure $(X, \leq, \sim)$ in which if action C corrupts data it increases entropy and so may be viewed as being evil. If C corrupts no data it leaves entropy invariant and so is viewed as benign.

Communications protocols are designed in layers of abstraction. When corrupt data are detected at a certain layer the next-highest layer regards them as having been lost and hence must cope with their retransmission. So we now consider the medium M, and hence action C, to be capable of losing data (but not of corrupting it).

The state of the system is similar with the exception that the sequence r of received data is a subsequence of s, the sequence of transmitted data:

$$X \ =_{def} \ \{(s, r) : D^* \times D^* \mid r \lhd s\},$$

where $r \lhd s$ means that r is a subsequence of s. The action of C can result in either accurate transmission

$$(s, r)\ C\ (s.d, r.d)$$

or loss

$$(s, r)\ C\ (s.d, r).$$

The level function is chosen to measure the number of lost items

$$\lambda(s, r) \ =_{def} \ \#s - \#r.$$

With respect to the pre-order defined by equation (1) an action which loses more data increases entropy more and so is more evil. A medium which loses no data leaves entropy invariant.

5.2 Virus

A virus typically deletes or otherwise corrupts files; less destructively, it may result in a message being displayed at some predetermined prompt. Let us see how to model all such eventualities at a given computer.

Since all those behaviours correspond to the computer not functioning as it should we choose for state space (of that part of Cyberspace in which we are interested) the set X of pairs (c, s), where c denotes the current program state of the computer and s denotes the state as specified: as it should be. An action of the computer relates state (c, s) to state (c', s') iff that action in fact transforms c to c' but ought to transform s to s'.

The level function we choose measures the deviation of the current state from its value as it ought to be. Thus $\lambda(c, s)$ equals the number of bits (say) in the program state not equal to their correct value (i.e. as determined by the second component of our system state). Even if the virus remains quiescent for some time after its invasion, it is still present as an incorrect component of the system's state and so is accurately captured by that definition of λ. The pre-order defined from that level function yields an entropy structure with respect to which a virus increases entropy. For the only way it can affect the computer is by modifying the computer's program state in some way.

Note that the function λ provides more detail than we need. Indeed even a single wrong bit in certain parts of a computer could have unpredictable consequences. So in this instance there seems little value attached to the ordering on actions introduced in section 4.2.

5.3 Webbot

In view of the huge amount of email broadcast indiscriminately to internet users, it is becoming popular to filter incoming email automatically. A webbot typically consists of a suite of programs that includes such filters. A webbot is faulty (evil) if it filters messages it should let pass, or conversely if it lets pass messages it should filter. The entropy method can be used to quantify the propensity of a webbot for evil as follows.

The state of the system in which we are interested consists, as in the previous two examples, of a pair. This time the set *in* represents the set of messages input by the webbot and the set *out* represents the set of messages passed on to the user. We assume that no new messages are introduced by the webbot; in other words *out* is a subset of *in*. Letting D denote the set of messages, the state space is

$$X \ =_{def} \ \{(in, out) : \mathbb{P}D \times \mathbb{P}D \mid out \subseteq in \wedge \#in < \infty\}$$

where $\mathbb{P}E$ denotes the set of all subsets of set E.

To define the level function it is convenient to introduce predicate *pass* on D defined so that $pass(d)$ is true iff message d should pass to the user. Then the set of messages handled incorrectly by the webbot is expressed

$$in \triangle out \ =_{def} \ \{d : out \mid \neg pass(d)\} \cup \{d : in \mid pass(d) \wedge d \notin out\}.$$

Now the level function is simply the size of that set

$$\lambda(in, out) \ =_{def} \ \#(in \triangle out).$$

A finer ordering is defined without use of a level function, directly in terms of the state,

$$(in_0, out_0) \le (in_1, out_1) \ =_{def} \ (in_0 \bigtriangleup out_0) \subseteq (in_1 \bigtriangleup out_1).$$

In either case we have an entropy structure with respect to which increase in entropy corresponds to evil action of the webbot. The correctly-functioning webbot leaves entropy invariant with $(in \bigtriangleup out) = \{\}$.

In a similar way we can model an *eavesdropper* able to monitor data passing along a communications medium. The system state is then modelled as the set *eve* of data eavesdropped and an action on the system is regarded as evil if it increases that set. So, as above, there are two choices of ordering. The first is defined by the level function

$$\lambda(eve) \ =_{def} \ \#eve$$

whilst the second pre-order is defined directly in terms of state

$$eve_0 \le eve_1 \ =_{def} \ eve_0 \subseteq eve_1.$$

According to the first (derived) pre-order, increase in entropy corresponds to a greater number of messages being eavesdropped. By the second, increase in entropy corresponds to an increase in the set of eavesdropped messages. But in both cases a 'benign' system leaves entropy invariant with $eve = \{\}$.

5.4 A faulty editor

In this section we demonstrate how a more complex entropy structure may be produced from simpler ones using the techniques of section 4.3. The example is that of a line editor, which inputs words and checks them for spelling against a dictionary.

Words are input from a keyboard and represented on screen by the editor. A *binding* associates each sequence of characters with its representation on screen (in practice that is done character-by-character but for this example we suppose it to be done by word). The correct binding is a function

$$c : W \longrightarrow Reps$$

where W denotes the set of all words (i.e. sequences of characters) and *Reps* denotes their representations on screen. The state of a line editor is a sequence of words input so far, with the representation of each

$$X_A \ =_{def} \ seq(W \times Reps).$$

Initially the state is empty. The action E_A of editing the next word extends the state by a pair consisting of the input word $\omega : W$ and its representation $\rho : Reps$

$$\sigma \, E_A \, \sigma.(\omega, \rho).$$

That action is deemed to be good or benign if all words are represented correctly and otherwise it is deemed to be evil. Such intuition is captured by taking the entropy structure defined by the level function $\lambda_A : X_A \longrightarrow \mathbb{N}$ which captures the number of words incorrectly represented

$$\lambda_A(\sigma) \;=_{def}\; \#\{i \mid c(p_1(\sigma(i))) \neq p_2(\sigma(i))\}$$

where projection functions p_1 and p_2 are defined by $p_1(\omega, \rho) = \omega$ and $p_2(\omega, \rho) = \rho$. The resulting entropy structure $\mathcal{A} =_{def} (X_A, \leq_A, \sim_A)$ provides a simple generic example.

Now suppose that it is deemed important to represent certain words correctly but that other words are of less concern. Perhaps, for example, correct representation of text is more important than that of strings of mathematical symbols; or *vice versa*. Such decisions are codified by a pre-order $\leq_W$ on words. For example to ensure that words in some set S are more important than others, take the pre-order given by the level function $\lambda_W : W \longrightarrow \mathbb{B}$, defined

$$\lambda_W(w) \;=_{def}\; (w \in S).$$

But of course in general pre-order $\leq_W$ may be total. To modify the entropy structure $\mathcal{A}$ to reflect that extra information, it suffices to use the lexical pre-order in which $\leq_A$ is refined by $\leq_W$. For then the resulting pre-order $\leq_B$ orders first states with no errors, then all states with one error, ordered where possible by $\leq_W$, then states with two errors and so on. The resulting entropy structure has the same state as $\mathcal{A}$, but the more complex lexical pre-order and equivalence defined in terms of simpler ones; we write it $\mathcal{B} =_{def} (X_A, \leq_B, \sim_B)$. So much for input of words.

A line editor uses a dictionary to check the spelling of the words it inputs. But the dictionary may have some errors, or might use an American dictionary to check British English text, or *vice versa*. Let $RH \subseteq W$ denote the actual dictionary used by the editor and $OED \subseteq W$ the correct one. (We overlook order in our dictionaries since we are here not interested in implementing the lookup function.) A word is correctly classified by dictionary RH iff both RH and OED would give the same answer, i.e. the word lies in the set

$$V \;=_{def}\; (RH \cap OED) \cup \overline{(RH \cup OED)},$$

where $\overline{S}$ denotes the complement of subset S of W. The edit action E_C appends to state a word and a bit saying whether or not that word is correctly classified by RH. Thus state is given by

$$X_C \;=_{def}\; seq(W \times \mathbb{B})$$

and the edit action given by

$$\sigma\, E_C\, (\sigma.\omega, (\omega \in V)).$$

A good or benign action appends only correct words; an evil one commits at least one error. That is captured by the entropy structure whose level function $\lambda_C : X_C \longrightarrow \mathbb{N}$ captures the number of words incorrectly spelt according to RH

$$\lambda_C(\sigma) = \#\{j \mid \neg p_2(\sigma(j))\}.$$

The result is another simple entropy structure $\mathcal{C} =_{def} (X_C, \leq_C, \sim_C)$.

Finally an editor uses a keyboard for entry of words and a dictionary to check the correctness of their spelling. Let us see how to define that simply in terms of $\mathcal{A}$ and $\mathcal{C}$. Firstly the state combines both X_A and X_C: it consists of a single sequence of input words

$$X_D \ =_{def} \ \{\sigma : X_A \times X_C \mid p_1(p_1(\sigma)) = p_1(p_2(\sigma))\}.$$

Action E_D extends state by a word together with its representation and the correctness of its spelling

$$\sigma\, E_C\, ((\sigma.\, \omega, \rho), (\sigma.\, \omega, (\omega \in V))).$$

Such an action ought to be good iff it is good for both components; similarly for benign actions; an action ought to be evil iff it is evil for at least one component. That is achieved precisely by the conjunction entropy structure $\mathcal{D} =_{def} (X_D, \leq_D, \sim_D)$ in which $\leq_D$ is the conjunction of pre-orders $\leq_A$ and $\leq_C$.

(As an exercise the reader may wish to define, solely in terms of the entropy structures introduced here and the combinators of section 4.3, an entropy structure which captures the decision that errors in representing a word on screen are worse than a faulty dictionary. Thus the ordering should be: first all states with no faulty word representations, ordered by the number of faults in RH, then all states with just one faulty representation, ordered by number of faults in RH, and so on.)

5.5 Breaking a windscreen

We now investigate what happens when the entropy method is applied to an example in the standard domain: one having no contact at all with Cyberspace. Since our method has been developed to deal with questions in Cyberspace, we expect this comparison to be revealing. Again, this example reiterates the importance of applying a hierarchy of entropy structures to analyse different viewpoints. Consider a child throwing a stone through a car windscreen at a tip. In its original form [5] the view is taken (a) that damaging (valuable) objects, even at the tip, is morally deprecable. Now we wish also to consider an alternative view: (b) that the destruction of an object may be beneficial with respect to the environment (recycling).

Corresponding to those two ethical stances are two entropy structures, each determined by a level function. Suppose that we wish to model a particular child throwing a particular stone against the windscreen of a particular car, and altering nothing else. Suppose furthermore that the action of the stone smashing the windscreen leaves the state of the stone unchanged whilst vastly altering the state of the windscreen. Let us define the system state to consist (in both cases) simply of the state of the windscreen. The two level functions are defined as follows.

(a) The intrinsic value of the windscreen is its ontic nature: the minimal condition of an object's least intrinsic worthiness can be identified with its abstract nature as an information entity [6]. That is, of course, determined by an ontological analysis. For the sake of simplicity, here we presume it to be determined by a function $\pounds$ which assigns to each windscreen state its commercial value (say a real number) $\pounds : E \longrightarrow \mathbb{R}$. We define the entropy structure to have pre-order $\leq_a$ equal to the converse of that derived from equation (1) using the level function $\pounds$. Thus by breaking the windscreen the child performs an entropy-increasing action.

(b) The extrinsic value of the windscreen is its latent energy. Thus by breaking the windscreen the child assists the process of recycling. That value is objective, determined say by the function $L : E \longrightarrow \mathbb{R}$. Again the entropy structure is derived from equation (1), this time using the level function L. Breaking the windscreen of the car is now an entropy-decreasing action.

That treatment is of course absurdly naïve. We have not allowed, for example, for the child's intent. Indeed were the breakage accidental we should wish to pass a different ethical judgement about it than were it deliberate. Similarly we have overlooked all other attributes of the state of mind of the child.

Note that the views (a) and (b) are opposite so of course one of them, (b), conflicts with the usual definition of entropy from thermodynamics. That reflects our freedom to choose entropy stuctures to suit our purpose.

6 Requirements revisited

In this section we consider the degree to which our proposal for Information Ethics is consistent with each of the requirements inntroduced in section 3.

Our proposal is evidently rigorous. The effect of an action on the pre-order of an entropy structure is a matter of proof or counterexample, working from the definition of the pre-order and specification of the action.

By being based on the mathematical properties of agents (that is, the specifications of their state spaces and actions) Information Ethics is stable: any changes to Information Technology, no matter how advanced, can nonetheless be described in mathematics.

Modularity has been demonstrated by the techniques for combining entropy structures. It, and the hierarchical reasoning it engenders, play an essential rôle in formal methods.

Finally we hope to have made a reasonable case for soundness by the variety of examples we have chosen.

By Computer Ethics most Computer Scientists mean Professional Ethics (for example [1], page 332). It is worth emphasising our view that Information Ethics provides a foundation for Computer Ethics which itself provides a domain for discussion of Professional Issues. Our proposal is not remote from application. Legal Issues of Cyberspace are particularly pressing. In a legal action an entropy structure could in principle be agreed by plaintiff and defendent, presumably the former seeking to strengthen and the latter to weaken its pre-order. Perhaps the judge would need to arbitrate to reach a compromise. Then the plaintiff would argue by example that the action involved increases entropy, whilst the defendent would argue by proof that it (preserves or) decreases it. Thus in principle our proposal aims to provide an appropriate foundation for issues as critical as legal issues.

7 Conclusion

Cyberspace consists of a world of beings which use their vast state spaces to perform complex and often far-reaching actions. The ethics of an action depends, amongst other things, on all its consequences (just as Consequentialism would argue in the standard case). But determination of those consequences is no more practically feasible in Cyberspace than it would in principle be in the standard world.

The approach of *standard Ethics* to the ethics of Cyberspace is to reason entirely outside Cyberspace, making whatever anthropic assumptions are necessary to do so. That has been found inappropriate and even misleading in its conclusions.

The *formal methods* approach is to reason rigorously from the specifications of all components involved to determine all consequences of a given action. Its advantage is that only with such methods can an exact understanding of actions in Cyberspace be completely understood. However, as we have observed, for large systems that is simply impractical: a system with two interacting components having p and q many states respectively has pq many states and so state explosion results.

We offer a simple alternative combining the beneficial aspects of both those approaches. Subjective judgement and consideration of common-sense values underlie the definition of a particular entropy structure. Rigorous reasoning is employed to determine how an action affects the entropy pre-order. The result seems to afford an exceptional simplification of the formal-methods approach by concentrating on just one aspect of an action.

Thus the contribution of this paper can be seen as a decomposition of the task of discussing the morality of an action in Cyberspace into standard and rigorous components.

References

[1] S. Baase. *A Gift of Fire: Social, Legal and Ethical Issues in Computing.* Prentice-Hall, 1997.

[2] T. W. Bynum and J. H. Moor, editors. *The Digital Phoenix: How Computers are Changing Philosophy.* Blackwell, 1998.

[3] B. A. Davey and H. A. Priestley. *Introduction to Lattices and Order*. Cambridge University Press, 1990.

[4] E. W. Dijkstra. How do we tell truths that might hurt? EWD498, 1975. In *Selected Writings on Computing: A Personal Perspective.* Springer Verlag, 1982, 129–131.

[5] L. Floridi. Information ethics: On the philosophical foundation of computer ethics. *Ethics and Information Technology*, **1**:37–56, 1999.

[6] L. Floridi. Does information have a moral worth in itself? In *Computer Ethics: Philosophical Enquiry (CEPE 98)*, edited by Lucas D. Introna, University of London Press, London, 1999.

[7] L. Floridi and J. W. Sanders. Artificial evil and the foundation of Computer Ethics. CEPE 2000.

[8] C. A. R. Hoare. An axiomatic basis for computer programming. *Communications of the ACM*, 576–583, October 1969.

[9] D. R. Hofstadter and D. C. Dennett. *The Mind's I.* Penguin Books Ltd., Marmondsworth Middlesex, 1981.

[10] Cliff B. Jones. *Systematic Software Development Using VDM.* Prentice-Hall International, 1986.

[11] T. M. Mitchell. *Machine Learning.* McGraw Hill, 1997.

[12] R. Motwani and P. Raghavan. *Randomized Algorithms.* Cambridge University Press, 1995.

[13] The Risks Digest, `http://catless.ncl.ac.uk/Risks`, the forum on risks to the public in computers and related systems organised by the ACM Committee on Computers and Public Policy.

[14] J. R. Searle. Minds, brains and programs. *Behavioural and Brain Sciences*, **3**:417–424, 1980. (Also chapter 22 in [9].)

[15] E. Steinhart. Digital metaphysics. In [2], 117–134.

[16] J. C. P. Woodcock and J. Davies. *Using Z: Specification, refinement and proof.* Prentice-Hall International, 1996.