8 Appendix 1 - A Brief Overview of Some Existing Formulations of Complexity

This is an overview of some of the articles which directly invoke the idea of complexity in their analysis, either by defining it or by specifying its properties. There is no comprehensive overview of this subject across disciplinary borders, but there are some relevant collections: [8, 14, 92, 159, 192, 346, 351, 422, 439, 475] and many more articles which include surveys *within* the bounds of individual subject areas: [14, 36, 57, 60, 70, 75, 79, 87, 94, 104, 106, 115, 217, 286, 295, 304, 355, 395, 397, 407, 414, 457, 477, 478, 498].

8.1 Abstract Computational Complexity

Blum [67] proposed an abstract definition of computational complexity. If $p_i(n)$ are the functions representing the computation of the programs P_i, then $c_i(n)$ are a set of complexity measures iff $c_i(n)$ is defined exactly when $p_i(n)$ is defined and the predicate $c_i(n)=m$ is decidable. This definition neatly includes the time and space measures as well as many other sensible resource measures (such as the number of jumps executed) and is strong enough to prove many of the important theories concerning them.

However this definition is too broad as it allows measures which don't obey the subprogram property (if P is a program that first applies a subprogram Q to an input and then a subprogram R to the result, then the complexity of P should be at least as great as that of Q or R). Thus according to this approach you get programs with more complex subprograms.

Fixes for the abstract definition of computational complexity are suggested by Turney in [444, 445, 446], and Ausiello suggests a weakened version in [31]. [66] argues that computational complexity should be extended over other fields like the real numbers.

8.2 Algorithmic Information Complexity

The Algorithmic Information Complexity (AIC) of a string of symbols is the length of the shortest program to produce it as an output. The program is usually taken as running on a Turing Machine. It was invented by Solomonoff [419], Kolmogorov [266] and Chaitin [99, 100, 101] separately, although perhaps anticipated by Vitushkin (section 8.45 on page 161). It has been one of the the most influential complexity measures (along with that of computational complexity) and has inspired many variations and enhancements including ‘sophistication’ (section 8.42 on page 159), and ‘logical depth’ (section 8.4 on page 138). Although Solomonoff considered it as a candidate for selection amongst equally supported scientific theories (i.e. a measure of simplicity – section 8.37 on page 156), Kolmogorov and Chaitin considered it as a measure of information (see section 8.15 on page 144 and section 8.24 on page 149).

It has many interesting formal properties [99], including:

1. *The more ordered the string, the shorter the program, and hence less complex.*
2. *Incompressible strings (those whose programs are not shorter than themselves) are indistinguishable from random strings.*
3. *Most long strings are incompressible.*
4. *In a range of formal systems you can't prove (within that system) that there are strings above a certain fixed level of complexity (derived basically from the AIC of its axioms).*
5. *In general it is uncomputable.*

Property 2 illustrates the deep connection between AIC and disorder. This is particularly evident in physics where a very close connection between Algorithmic complexity and entropy has been shown [449], to the extent that it is often referred to as an entropy.

Property 4 indicates that the AIC complexity is more of an information measure. While one might believe that it is not possible to produce more information within a formal system than is encoded by the axioms, it would be extremely counter-intuitive if there was a limit to how complex one could prove strings in it to be.

AIC has been applied in many ways: to define randomness in a non-probabilistic way [309, 499]; to capture descriptive complexity [293] (see also section 8.8 on page 141); Rissanen uses a statistical version to motivate a principled trade-off between the size of model and its error in [377, 378, 379]; to biological complexity [203, 223, 345]; to cognitive models [397]; economic models [452] and data compression [499]. Lempel-Ziv encoding can be seen as a computable approximation to it [279, 495].

Given that it is better characterised as an information measure rather than complexity, it has very close connections with entropy, as explored in [169, 420, 496, 497]. It is generalised in [80].

Good summaries of the many formal results and applications can be found in [40, 41, 70, 118, 286]. Other formal results include [80, 284, 456, 451, 494]. A summary of philosophical applications can be found in [285], with others in [102, 301, 469].

For more discussion on this see section 4.3.4 on page 84 and section 3.4.1 on page 57.

8.3 Arithmetic Complexity

This is the minimum number of arithmetic operations needed to complete a task. This is important in order to make computational algorithms more efficient, for example Strassen [424] improved upon Gauss's method for solving linear equations from $O(n^3)$ operations to $c \times n^{2.71}$.

This is more of a practical definition and not intended as a general model of complexity. The operations of arithmetic are very particular. It also does not take into account the precision of the operations or of rounding errors. A summary of the theory of the arithmetic hierarchy can be found in [178].

8.4 Bennett's ‘Logical Depth’

Bennett [54, 55, 56] defines ‘logical depth’ as the running-time to generate the object in question by a near-incompressible program. Strictly the depth of a string x at level s is: $D_s(x) = min\{T(p) \,|\, |p| - |p^*| < s \wedge U(p) = x\}$, where p ranges over programs, *T(p)* is the time taken by program *p*, *p** is the smallest such program and *U* is a universal Turing computer.

He states that this is intended as a measure of the value of information. For example, tide-tables can have a greater value than the equations which were used to calculate them as a lot of useful computation has been done. Thus he says [57]:

> *“Logically deep objects… contain internal evidence of having been the result of a long computation or slow-to-simulate dynamically process and could not plausibly have originated otherwise.”*.

The plausibility of its origin comes from the assumption that the most likely program to produce an output would be the shortest one. This idea comes from Solomonoff.

He justifies this as a physically plausible measure of complexity by its obedience to the “slow growth law” of complexity. This informal law states that complexity can only arise slowly through stochastic processes, as presumably has occurred in evolution. By its construction one cannot produce a deep object from a shallow one by a deterministic process and only improbably by a stochastic one.

Thus random strings and very simple ones both have a low logical depth. A random string is incompressible and hence the minimal program that produces it, is a simple copying program, which is quick. A simple pattern can be produced by a simple program, and so will also be fairly quick.

Koppel [268] shows that Logical Depth is the same as Sophistication (section 8.42 on page 159) for infinite strings.

8.5 Cognitive Complexity

In cognitive psychology, several types of complexity are distinguished. The most discussed of these is Cognitive Complexity. This was defined by Kelly as a part of his theory of personality [250]. He developed his ‘role construct repertory’ test to test it. Since then it has been used as a basis for discussion on the complexity of personal constructions of the real world (and particularly of other people) in psychology. It asks the subjects to rate a number of people known to them (e.g. closest friend of same sex) on a number of attributes (like Outgoing vs. Shy). The dimension of the inferred mental model of these people is then estimated as their cognitive complexity.

So, for example, people who assign to all their friends positive attributes and to their enemies negative attributes would have a one-dimensional mental model of their acquaintances, as everybody is aligned along this good/friend - bad/enemy scale. Such people are said to be “cognitively simple”. A person who indicated that some of both their friends and enemies were good and bad would have at least a two-dimensional model with people placed across a good-bad, friend-enemy pair of axes. This person would have a higher score and would be called “more cognitively complex”. Thus the level of cognitive complexity indicates the number of potential relationships between the various attributes.

Quite a number of variations of this has been suggested to capture this idea [404]. Unfortunately these seem to measure slightly different things as they do not correlate in practice [204], although they do have some robustness over time [342]. There does not seem to be any strong connection between cognitive complexity and IQ [95], innovation [187], intellectual sophistication [416], loquacity [81] or educational level [366]. It does seem to have some relation to the ability to use complex language [48, 409]. The application of hierarchically structured algorithmic information is discussed in [397]. A synthesis of several measures of cognitive complexity is suggested in [413] in the internal representation used by subjects.

Other related measures include: [59, 382, 426].

8.6 Connectivity

The greater the extent of inter-connections between components of a system, the more difficult it is to decompose the system without changing its behaviour. Thus the connectance of a system (especially when analysed as a graph [367]) becomes a good indication of the potential for complex behaviour, in particular the likelihood that the system will achieve an equilibrium. The connectivity of a system has been variously measured, including the number of relations (section 8.30 on page 153) and the cyclomatic number (section 8.7 on page 140).

Applications include: the reliability of circuits [470]; the stability of random linear systems of equations [25]; stability in computational communities [259]; stability in ecosystems [86, 227, 353]; the diversity of ecosystems [308]; the structure of memory [273]; logical and computational properties of bounded graphs [319]; competition in networks [373]; random digraphs [405]; chemical reaction mechanisms [491]; and general emergent behaviour in biological systems [197].

8.7 Cyclomatic Number

The most basic graph measure (apart from the number of vertices) is the cyclomatic number of the graph. This is basically the number of independent loops in a graph. It is easily calculated by the formula $\nu(G) = m - n + p$, where m is the number of arcs, n the number of vertices and p the number of disjoint partitions the graph divides into.

This intuitively captures the inter-connectedness of a graph or system; a hierarchically structured machine is completely predictable (a tree has no loops), whilst one with many feedback loops can exhibit more complex behaviour. An army is organised on hierarchical lines, presumably to simplify the chain of command to make it more predictable and hence more controllable. On the other hand, a creative committee meets to allow the maximum number of communication channels to enable the unpredictable to occur.

In general there is no direct relation between the size (number of nodes) and the (cyclomatic) complexity. If a system is represented by a graph with the presence of some relation indicated by an arc, then the number of nodes will limit the cyclomatic complexity. This effect is only significant with very few nodes as the number of possible arcs goes up exponentially with the number of nodes. For the theory of this area see [436]

McCabe [313] uses this as a measure of program complexity, in particular to calculate the number of different logical paths through a program to gauge how many tests it might need. Other applications include: complexity of simulation models [403]; and the difficulty of software maintenance [49, 125, 232].

For discussion on this see section 5.4 on page 106.

8.8 Descriptive/Interpretative Complexity

Löfgren [293] writing from a biological and psychological context, distinguishes between descriptive and interpretative complexities. In a system with a description (like DNA) and its realisation (the proteins in the cell), he associates his two measures of complexity with the two processes of interpretation and description. That is the complexity of encoding the realisation into a descriptive code and decoding it back into a realisation of that code.

Löfgren chooses Kolmogorov complexity (section 8.2 on page 136) for the process of description and an ordering based on logical strength (section 8.20 on page 146) for the interpretative complexity.

8.9 Dimension of Attractor

Chaotic processes are difficult to model. A small change in state now causes a large change later which makes it impossible to predict the exact state beyond a certain time limit. This does not mean that all aspects of the process are impossible to model. It is possible to estimate the processes' attractor in state space; this is often fractal with chaotic processes. The dimension of the attractor is a measure of how complex the process is.

Depending on the method of convergence for the calculation of a dimension for the attractor you get a slightly different measure. These, in fact, form a sequence of dimensions. For an accessible introduction see Baker [39].

8.10 Ease of Decomposition

The ease with which a system can be decomposed into sub-systems has close connections with the “analytic complexity” of section 5.2 on page 87. The general area is covered by [18, 110, 144, 338, 421]. Some techniques for systematic decomposition are: the use of a matrix algorithm to plan the use of multiplexers for circuits [278]; a graphical approach in [242]; decomposing difference equations [30, 306]; a hierarchical holographic algorithm [72]; the design of decision support facilities [225]; a systems approach [157]; and a technique based on whether data relations commute [131].

The converse of and complement to decomposability is reconstructability analysis [94].

8.11 Economic Complexity

“Complexity” in economics, frequently means merely that some of the usual simplifying assumptions do not hold. An example of these assumptions is that an agent acts as if it can infer the action to perfectly optimise its utility. This goes back to Simon’s distinction between procedural and substantive rationality [415]. See the paper in Appendix 7 - Complexity and Economics, for a full discussion of the concept of complexity in economics. Some papers that cover this are [10, 14, 20, 172, 173, 200].

In game-theory, there has been some more direct formulation of actual complexity measures, including: a critique of the “number of states” measure [43] (section 8.32 on page 153); the information of strategies [290]; a survey of the area [229].

Another area deals with choice processes, including: the group-theoretic complexity of decision rules [163]; a survey of choice processes and complexity [192], the computability of choice functions [251]; hierarchies [484]; and the cardinality of collections of decisions [52].

8.12 Entropy

In physics, entropy measure the level of disorder in a thermodynamic system. The more disordered it is, the more information is needed to describe it precisely. In particular systems with very low entropy are simple to describe (they don't move around a lot). Thus complexity and entropy can be associated, although this was not intended by its originators [408]. Entropy based measures are essentially probabilistic. The Boltzman-Gibbs-Shannon entropy is most frequently used in physics, but Algorithmic Complexity can also be used if the complexity of the whole ensemble is low [497].

The principle of maximum entropy [282] has been used to help formalise complexity [114, 156, 171].

Entropy based measures have often been used as measures of complexity including: the regularity in noisy time series [354]; the topology of chemical reactions [492]; coalitions of economic agents [452]; physical computation [496]; the difficulty of system diagnosis [183]; artificial life [371]; and the complexity of graphs [332].

8.13 Goodman's Complexity

Goodman [186] has devised an elaborate categorisation of extra-logical predicates, based on expressiveness. For example, a general predicate is deemed more complex than a symmetric one, as it includes the later as a specific example. Likewise a three place predicate is more complex than a two place one. Goodman builds upon this starting point. The idea is that when faced with two theories that have equal supporting experimental evidence one should choose the simpler one using this measure.

The complexity of a complex statement is merely the sum of the complexities of its component predicates, regardless of the structure of the statement. It is similar in spirit to Kemeny’s measure (section 8.18 on page 146). A recent defence and reformulation of this idea has been made by Richmond in [376].

8.14 Horn Complexity

The Horn complexity of a propositional function is the minimum length of a Horn formula (in its working variables) that defines that function. This was defined by Aanderaa and Börger [1] as a measure of the logical complexity of Boolean functions. It is polynomially related to the network complexity [2], described below (section 8.26 on page 151).

8.15 Information

The amount of information a system encodes or the amount of information needed to describe a system has a loose connection with its complexity. As noted above, there is a close connection between the amount of information and disorder. Using the Algorithmic Complexity (section 8.2 on page 136) measure of information, disordered patterns hold the most information, patterns encoding the maximum amount of information are indistinguishable from random patterns.

Information can be measured deterministically using algorithmic information complexity (section 8.2 on page 136) or probabilistically using entropy (section 8.12 on page 143). Either of these can be used to define mutual information (section 8.25 on page 151). See also section immediately below (section 8.16 on page 145).

Klir exhibits an axiomatic framework for complexity similar to those I list in section 5.2 on page 87, combined with the requirement that complexity should be proportional to the information required to resolve any uncertainty [262, 263, 264, 265]. This may be seen as a formulation of Waxman's "problem complexity" [463].

A number of approaches which seek to combine elements of both algorithmic and shannon information include: [170, 378, 452, 496].

Computational complexity has been extended to cover information flow by adding a cost function to the information used by a computation [440, 441, 442].

Applications include: charting the increase in information in the evolution of finite automata [26, 27]; the fluctuation of information in 1-D automata [47]; its connection with logical depth in evolution [53]; its connections to computational complexity [138]; the connection between various measures of information via random vectors [144]; the regularity of short noisy chaotic series [160]; error-prone sections of programs by potential information flow [214, 277]; systems problem solving [262]; the classification of strategies in repeated games [290]; the estimation of the information of a pattern [340]; and a principle of the minimum increase in evolution [399]

8.16 Information Gain in Hierarchically Approximation and Scaling

Attention in physics has focused on the complexity of chaotic physical processes with a fractal nature, where one gets different behaviours at different levels of granularity. In 1986 Grassberger introduced “Effective Measure Complexity” [193], which measured the asymptotic increase in information with increased scale. He develops this in [194, 195].

Badii, Politi and others [33, 34, 35, 36, 130] use trees of increasingly detailed Markov models to approximate a growing pattern. Each branch off a node is a possible extension of the pattern that may follow. He then defines the complexity of the pattern as the (Shannon) information gain in each level over the size of the tree at the level, taking the limit at infinitely many levels. Any Markov process has zero complexity. This is to reflect the difficulty in predicting complex systems. The class of easily predicted systems that Badii focuses on are those which exhibit different behaviours at different levels of detail. He says

> *“A system is complex if it reveals different laws (interactions) at different resolution (coarse grinning) levels.”*.

[202] argues that these measures assume that the process is stationary, i.e. is basically a Markov process and [457] classifies them according to whether they are based on homogeneous or generated partitions and whether they are based on dynamic or structural elements. Other papers in this area include: [3, 7, 28, 29, 350, 480, 490]. A good review of this whole area is [36].

8.17 Irreducibility

Holists often use the word “complexity” for that which is irreducible [339] (at least by current practice). This is, in a sense, an extreme case of the difficulty of decomposition (section 8.10 on page 142). Such approaches include: [13] where the importance of size to qualitative behaviour is pointed out; [468] which argues that the evolution of multiple and overlapping functions will limit reduction in biology; [11] which discusses the application to public policy in forestry; [247] which charts how chaos challenges the reductionist approach; [257] which applies this to modelling organisations; as a result of self-organisation [205]; the incompatibility of information and computation [243]; as a result of the epistemic cut between syntax and semantics [347]; number of elements an instance of a pattern must consist of to exhibit all the characteristics of a class [210]; and [323] which discusses Rosen’s approach [384, 389] and relates this to the “sciences of complexity”. This approach to complexity seems particular to biology, for general surveys of the connection of complexity with holism see [244, 348, 388, 481].

Some suggest that this may be due to using the wrong formal language for modelling, including [74, 167, 246, 244, 330, 344, 410, 482, 488].

8.18 Kemeny's Complexity

In the field of “simplicity”, Kemeny [254] attributes an integral measure of complexity to types of extra-logical predicates. He does it on the basis on the logarithm of the number of non-isomorphic finite models a predicate type has. On the basis of this he gives extra-logical predicates a complexity which could be used to decide between equally supported theories. This is similar in style and direction to Goodman’s measure in section 8.13 on page 143.

8.19 Length of Proof

Simpler theorems, on the whole, need shorter proofs. On the other hand longer proofs are tedious to follow. Thus it is natural to search for short proofs (e.g. as in [96]).

One can arbitrarily lengthen almost any proof. This alone makes length alone as a measure of complexity unsatisfactory. Some short “elegant” proofs are very complicated and some careful long explanatory proofs easy to follow. For this measure to make any sense needless length needs to be eliminated (see minimum size measures in section 8.24 on page 149). Other papers touching on this include: [180, 209].

8.20 Logical Complexity/Arithmetic Hierarchy

Mathematical proof theorists classify mathematical objects and processes according to the projective hierarchy (sometimes called the arithmetic hierarchy). If the definition of an object is logically equivalent to a statement with alternating quantifiers:

$$\forall x_1 \exists x_2 \forall x_3 \exists x_4 \ldots Q x_n R [x_1, \ldots, x_n]$$

where Q is a quantifier and R is a quantifier free logical proposition in the variables $x_1,\ldots,x_n$ then the object is said to be a member of the class Π^1_n.

Likewise if the statement is of the form:

$\exists x_1 \forall x_2 \exists x_3 \forall x_4 \ldots Q x_n R [x_1, \ldots, x_n]$

then it is in the class Σ^1_n.

If a statement is provably equivalent to both a Σ^1_n statement and a Π^1_n statement it is called a Δ^1_n statement. The top numeral refers to the type of function or object allowed in the statement. The whole hierarchy looks like figure 28 (inclusions go upwards).

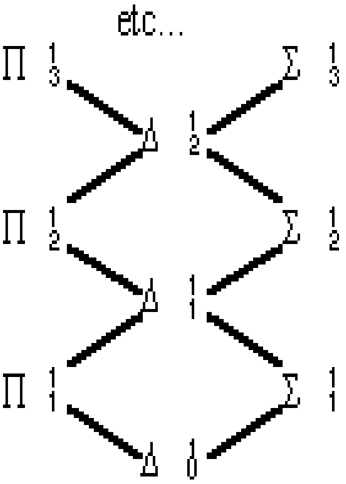

Figure 28. The arithmetic complexity hierarchy

Basically, as one ascends the hierarchy the statements in the classes can have more expressive power and they are more difficult to prove or model (in the mathematical sense). Girard [178] surveys this area thoroughly. [107] shows that the arithmetic complexity of the problem of deriving a word from a fixed starting point is arbitrarily more complicated than the word problem itself.

8.21 Loop Complexity

The loop complexity of a primitive recursive function is the iteration depth of the primitive recursive register operators in its definition. Thus x+1 would be level 0, x+y level 1 (as it can be defined recursively from x+1), x×y level 2 etc. This can be used to define a hierarchy of sets $LOOP_n$ of functions with loop complexity not greater than *n* [322]. This is one of the large collection of measures used to predict the maintainability of software, for a survey of these see [498].

8.22 Low Probability

The connection of probability and complexity is intricate. The probability of a highly ordered complex system arising by chance is low, hence sometimes complexity is associated with low probability [136]. On the other hand if complexity is conflated with informational measures such as entropy (section 8.12 on page 143) or algorithmic information (section 8.2 on page 136) then complexity is associated with high probability. This has led many to look for other measures such that complex systems will lie between order and disorder including those in section 8.16 on page 145 and section 8.46 on page 161.

In the contrary direction the philosophy of “simplicity” (section 6.5 on page 129 and section 8.37 on page 156) has lead to the identification of a higher a priori probability of the truth of a theory with a lack of complexity. That this is mistaken see the arguments in section 6.5 on page 129 and the paper in Appendix 6 - Complexity and Scientific Modelling.

An application of low probability to the difficulty of system diagnosis is in [183].

8.23 Minimum Number of Sub Groups

The Krohn-Rhodes prime decomposition theory [17, 272] tells us that we can decompose any semi-group in to a wreath product of alternating simple groups and semi-groups of order 3. There are decompositions which are minimal in terms of the number of such alternations, i.e. they have the least number of groups in their decomposition. The number of such groups is called the complexity of the original group. If you take the product of a group with another group then the result will be more complex, which accords with our intuitions.

Gottinger [190] derives this measure from three “Axioms of Complexity”, which I reproduce below, using the author’s system of notation ($\theta : M_f \to Z$ is the complexity function). He is writing in the context of considering the semi-group of the transformations between states of a machine.

1.a) $f|g$ implies $\theta(f) \leq \theta(g)$

b) $\theta(f_1 \oplus \ldots \oplus f_n) = max\{\theta(f_i) | i = 1, \ldots, n\}$

2. For all machines $f_1, f_2 \in M_f$

$\theta(f_1 \otimes f_2) \leq \theta(f_1) + \theta(f_2)$

If there is a feedback operation » from f_1 to f_2, then

$\theta(f_1 \otimes f_2) \leq \theta(f_1) + \theta(f_2) + \theta(f_1 \gg f_2)$

3. $\theta(U_3^f) = 0, \theta(D_1) = 0.$

Here: U_3^f is the semi-group for a flip flop operations and D_1 is a simple delay.

Axiom 1a) is the familiar subsystem property, saying that subgroups (strictly homomorphic images of groups) are no more complex that their parent group. Axiom 2 asserts that, if there is no “feedback” between two semi-groups, the complexity of their parallel composition is not more than the sum of their complexities. If there is a feedback operation then the complexity of their parallel composition is not more than the sum of their complexities plus the sum of the complexity of the feedback operation. Axiom 3 states that certain very small semi-groups have zero complexity. These are believable in terms of our intuitions about complexity (even if not all all of them are obvious).

Axiom 1b) is not so acceptable. It states that the complexity of a serial composition of sub-groups is only as complex as its most complex component, however complex or numerous the other sub-groups might be. All the other sub-groups in the serial composition make no difference to the overall complexity and new ones could be added ad nauseam! It is this axiom which makes this characterisation of complexity one of a maximal nature.

The name “simple group” is very misleading. Simple groups are merely groups that cannot be decomposed into smaller groups. The commutative (abelian) groups are well understood and have a high degree of symmetry. On the other hand some non-abelian groups are far from simple! Some are so big and complex that they have be renamed as the “monster” groups. All these groups will have the lowest possible complexity by this measure.

Given a set of groups, all of the same size, then the more a group decomposes into small groups the more complex it is deemed to be. A group that hardly decomposes at all, where we are left with little easy structural analysis, is deemed simple.

This has been applied to economic and social systems in [10, 189, 191, 163].

8.24 Minimum Size

As discussed in section 3.4.3 on page 58, minimum size overcomes some of the inadequacies of mere size as a complexity measure. It avoids the possibility of needless length and is nicely independent of the particular expression chosen. It would correspond to using a perfectly efficient language, the occurrence of any redundancy in a specific expression was eliminated by perfect compression.

However the minimum size of a particular representation can still be a largely accidental feature of the description process. Different ideas are sometimes more succinctly expressed in different languages (national and formal). For example, to express a conjunction in a negation-implication fragment of classical propositional logic is necessarily longer than that for implication itself. This would not mean that implication was simpler that conjunction.

Minimum size also ignores any question of inter-relatedness or relevance. Compare the cases of 1001 inter-related facts about logic and 1001 unrelated general knowledge facts (presuming this to be possible). It is probably possible to compress the 1001 inter-related facts more than the unrelated ones because the very fact of their relatedness indicates a degree of redundancy. The minimum size approach to complexity would thus attribute a lower complexity to the 1001 related facts but few would say these were less complex. What is true is that the system of unrelated facts holds more information but is far complex (as a system).

In a way that is similar to what occurred for length measures, some of these seem to have had the label “complexity” applied post-hoc, so it is difficult to judge how seriously they were meant as a complexity measure.

The most frequent special case of a minimum size measure is Algorithmic Information which corresponds to a minimal sized Turing Machine (see section 8.2 on page 136), but other minimum size approaches are also used with other formal languages, in particular finite automata (see section 8.32 on page 153), this approach is criticised in [43] for the analysis of equilibria in repeated games.

Crutchfield generalises the minimal size criterion over the whole formal language hierarchy, so that complexity is the minimal size in the “lowest” formal language for which this is finite [121]. He contrasts this complexity measure with a version of effective measure complexity (section 8.16 on page 145) which he calls “statistical complexity” [123]. The method for finding such an expression is given in [119] and applied to the process of modelling chaotic process in [120, 121].

Minimum size measures have also been applied to capture the static complexity of cellular automata in [472]; and to a minimal complexity in evolution [296].

8.25 Mutual Information

If you have defined an entropy like measure (e.g. Shannon Entropy or Algorithmic Complexity), H(A) and from that a joint entropy H(A;B) which is the entropy of A and B joined, then you can define. This can be interpreted as, i.e. the extent of the shortening when considered together rather than separately.

A high mutual information between remote parts of a system can indicate a closely connected or self-similar system. The connectiveness in such a system can be the cause of its complexity. Bennett [55] points out that this arises for rather different reason in equilibrium and non-equilibrium systems. In equilibrium situations the mutual information comes from the intervening medium (like in a gas), in non-equilibrium systems it must come from some other connection. He points out that simple operations like duplicating and mixing up random bits of DNA generate large amount of remote non-equilibrium mutual information.

[287] shows that past-future mutual entropy is not related to entropy in a straight-forward manner. [7] formulates “physical complexity” as the mutual information (defined relative to a Landuer-Turing Machine) between a systems and its universe. Mutual information has been applied to capture some of the dynamic complexity of cellular automata in [289].

8.26 Network Complexity

Network or circuit complexity is the minimum number of logical gates needed to implement a logical function [400]. This is very difficult to compute in most cases but some upper and lower limits can be proved. This measure depends on the choice of logic gates that you can use to build the circuits from.

This measure has an immediate importance for electronic engineers who seek to minimise the expense of logic gates as in [278]. This is polynomially related to Horn Complexity (section 8.14 on page 143). For surveys of results in this field see [40, 142, 400].

8.27 Number of Axioms

Meredith and Lukasiewics both put considerable effort into finding small axiom sets for classical propositional logic. For example Lukasiewics [300] proved that

$$[\,(p \to q) \to r] \to [\,(r \to p) \to (s \to p)\,]$$

was the shortest possible single axiom for the pure implicational propositional calculus. However it is far from obvious that this is helpful. The axiom has no immediately comprehensible meaning and it makes for an incredibly tortuous proof theory. For more on this see section 5.6.1 on page 113.

8.28 Number of Dimensions

In any model of a process, the number of dimensions it takes is of critical importance. A necessarily high dimensional model has the potential for great complexity. Conversely if there is a simple relationship between dimensions in a model you can often reduce the models dimension by forming composite dimensions with out any loss of descriptive power. Hence if a model is necessarily of high dimension then there is no very simple relationship between any of its several dimensions, i.e. the model must be reasonably complex.

This has been applied to concept learning [298]; the performance of connecting networks [307] and in cognitive complexity (section 8.5 on page 139). Fractal dimension is used to measure plant development in [113].

8.29 Number of Inequivalent Descriptions

If a system can be modelled in many different and irreconcilable ways, then we will always have to settle for an incomplete model of that system. In such circumstances the system may well exhibit behaviour that would only be predicted by another model. Thus such systems are, in a fundamental way, irreducible. Thus the presence of multiple inequivalent models are considered by some as the key characteristic of “complexity”. These people are usually holists, namely [323, 389]. See also section 4.3.1 on page 83.

This approach can be extended in restricted circumstances to measuring complexity by the number of inequivalent descriptions [88, 89].

8.30 Number of Internal Relations

If one is focusing on the topology of a model, then one improvement on the simple size of the network as an indication of its complexity is the number of relations indicated between the nodes.

Rouse and Rouse [392] in their study of the time taken to complete tasks found a strong correlation between the time taken to perform fault diagnosis tasks with complex relations and the number of internal relations in that circuit (represented by a wiring connection). Van Emden [450] examines the mathematics of a variety of entropic measures based on the information indicated by the internal relations at different levels.

8.31 Number of Spanning Trees

An interesting graphical measure is the number of spanning trees of a graph (see also section 8.7 on page 140). A spanning tree is a subgraph with no loops which includes all the vertices. The number of spanning trees grows very fast with the cyclomatic number and size of the graph. A tree has only one spanning tree [252]. [58] applies this to a classification of games. [226, 237] use such trees as the basis for a measure of complexity to capture the variety in the structure of trees.

8.32 Number of States in a Finite Automata

Much formal work [231] has been done on the number of states of finite automata. In these works this number is frequently taken as the complexity (e.g. [164]). Again it is easy to elaborate a model by adding redundant states, a difficulty which is circumvented by selecting a minimal or "acceptable" model (see section 8.24 on page 149 above).

Gaines [165] is pessimistic about a useful general theory of complexity, saying:

> "*The ordering of models in terms of complexity is arbitrary and depends upon our individual points of view.*",

and again:

> "*When we specify an order relation upon the models we may find that the behaviours of many important systems require complex models under our ordering, whereas, with a different ordering on the same class of models, they all become simple.*".

He, nonetheless, introduces the useful concept of admissibility (borrowed from statistics [465]) and applies it to the search for simple finite-state automata for various string patterns. He uses a working definition of complexity which counts the number of states of an automaton and then goes on to identify, with the help of the program ATOM, admissible models of various sizes. Here models are said to be admissible if any other model that gives a better approximation of the behaviour is more complex (in the sense of number of states).

In [164] he shows that even a small amount of randomness can cause an indefinite increase in an induced automata model. This work is extended in [396] to stochastic automata.

Von Neuman speculated that there was a critical threshold which allowed self-reproduction [454]. In [303] it is shown that Turing machines with very few states can exhibit complex behaviour.

Complexity as the number of states in a finite automata has been widely applied: to characterise the emerging complexity resulting from the actions of cellular automata [289, 472, 473, 474, 476]; in economic game theory [229]; to characterising social structure [9] and in characterising the computation done in chaotic systems [122].

8.33 Number of Symbols

The number of symbols is not a reliable guide to complexity. Merely to count the number of symbols in philosophical works would give one little indication of their complexity. Also compare the following logical statements:

$$(6) \quad (a \to b) \to (a \to b)$$

and

$$(7) \quad p \to (p \to p) \ .$$

Under almost any length measure the first is more complex than the second, yet intuitively a trivial instance of identity is less complex than the troublesome mingle axiom.

You do need a certain number of symbols for expressive power. Jaskowski [240] proved that you need at least one axiom with eleven symbols or two with nine in an axiomatisation of Classical Logic.

Size does create resource problems and hence needless size is undesirable. This is especially true for us humans who have a distinct limit on our short term memory. So, things that overload our short term memory can be difficult to understand. This is completely different situation from that where we can readily hold the information in our head but find it difficult to comprehend. We can deal with first difficulty given enough paper and time, the second is not necessarily any easier when written down.

This measure is used most frequently in linguistics (e.g. [45, 153, 256, 305].

8.34 Number of Variables

The number of variables in a statement can have an immediate impact both on proofs that use it and the complexity of its models. Both of these effects depend on the structure of the statement. For example the axiom $a \to b$ has a catastrophically simplifying effect on both proofs and models compared to that of $a \to a$.

As with the number of symbols (section 8.33 on page 154 above) the number of variables can have a limiting effect on complexity but the number of variables is not a sufficient condition for complexity. Diamond and McKinsey proved [139] that for a broad range of logics you need at least one axiom with three variables in it.

8.35 Organised/Disorganised Complexity

Weaver [464] classified scientific problems into the simple, and the complex.Then he further classified the complex problems into those of disorganised complexity and organised complexity. Simple problems are those with a few variables like the path of a billiard ball and a complex problem is one with many variables like a gas. Disorganised complexity is typified by many independent variables, so that it is amenable to statistical techniques. Examples of this are the properties of a gas or a nation's accident statistics. Organised complexity occurs when *“There is a sizeable number of factors which are interrelated into an organic whole”* Examples given by him include the immune system of animals and economic fluctuations.

8.36 Shannon Information

Although Shannon [408] did not envisage his measure of information being used to quantify complexity, some of his successors have either used it as such or based complexity measures upon it.

The Shannon measure of information is a statistical measure based on the probability of receiving a message. If $p(m_1), p(m_2), \ldots$ are the probabilities of receiving the messages $m_1, m_2, \ldots$ then the information carried by the message $n_1, n_2, \ldots$ is defined as $-\sum_i \log_2(p(n_i))$. The more improbable the message, the more information it gives the recipient.

See the section on information (section 8.15 on page 144) and entropy (section 8.12 on page 143).

8.37 Simplicity

When faced with two theories which are equally supported by the available experimental evidence, it is natural to choose the simpler of the two. Further than this, when a theory has been elaborated in order to explain the evidence, it is often fruitful to search for a simpler theory. The study of the grounds for choosing between equally supported theories has acquired the label "Simplicity" (see [4, 78, 186, 253, 254, 339, 485]).

From the point of view of theories about the world, all purely logical propositions are equally and ultimately certain and hence "simple". Thus measures of simplicity do not help us to distinguish between logical theories, they were not meant to. Many theories of Simplicity have chosen grounds other than simplicity as the criterion for choosing between equally supported theories, e.g. Popper's refutability [358] or Defrays [241] identification of Simplicity with high probability. Some theories with connections with complexity are Goodman's (section 8.13 on page 143), Kemeny's (section 8.18 on page 146) and Sobers (section 8.41 on page 158).

For a fuller discussion of this see section 6.5 on page 129 and Appendix 6 - Complexity and Scientific Modelling.

8.38 Size

There is clearly a sense in which people use "complexity" to indicate the number of parts but seems rarely used just to indicate this. It would be odd for a person opening a phone book or a large box of matches to exclaim "Oh, how complex!". Contrast these examples with those of a mathematical text book or an intricate (old fashioned) watch, where this would be more appropriate. Size seems not to be a sufficient condition for complexity.

On the other hand a certain minimum size does seem to be a necessary condition for complexity. It is very hard to imagine anything complex made of only two parts. However, this minimum size can be quite small: small non-abelian mathematical groups can be very complex indeed as are many other formal systems with a sparse axiomatisation. The rate of potential complexity seems to increase very fast with size. This does not, of course, mean that all large systems are complex.

Size based measures of complexity seem to come about in two circumstances: as a result of a post-hoc labelling of a formal device (as in simple induction proofs where the length of a proof, the number of connectives or the depth of nesting is in need of a convenient label) and to indicate a potential for complexity (as in the number of variables in a formula).

Anderson points out that size can make a qualitative difference to the behaviour of systems [13] as [454] also suggests, but [303] indicates that in the presence of powerful inferential machinery that the critical size can be very small.

Applications include: the social organisation and community size [83]; the minimum number of gates in a circuit [278]; the cyclical behaviour of systems [458]; self-replicating sequences [44]; rule-based systems [341]; neural networks and cellular automata [188]; and grammatical development [256].

See also the discussion in section 3.4.1 on page 57 and the other size and numerosity based approaches in this appendix.

8.39 Size of Grammar

A pattern, if viewed as the result of production rules in a language, has a grammar [231]. In general the simpler the pattern, the simpler the grammar. So the size of the grammar gives us a handle on the complexity of the pattern. The size and complexity of the grammar can vary depending on what sort of language you are assuming the pattern to be a representative of. For instance Gaines [164] shows that the assumption that a process can be modelled by a deterministic finite automaton leads to very large models (proportional to the length of the evidence) in the presence of even a small amount of indeterminism. This measure would identify all patterns of a particular language as equally complex unless the pattern happens also to be a member of another language as well. Sahal [396] demonstrates similar results, but with stochastic automata.

Frequently the size of grammar is taken relative to a Turing machine (section 8.2 on page 136) or finite automata (section 8.32 on page 153). Other approaches include simple depth (section 8.44 on page 160) or star height in regular languages [153].

Applications include: biological macromolecules [146]; chaotic systems in physics (section 8.16 on page 145); and communication complexity [235].

8.40 Size of matrix

The size of a minimal characteristic matrix for a logic is an indication of the logic's complexity [209]. Classical logic has the smallest possible matrices (2x2), and more complex logics like R, do not have finite characteristic matrices at all. This measure is an indication of the the complexity of logic's semantics but does not have a direct relationship with the complexity of its proof theory (see section 5.6.2 on page 117).

This sort of approach has been applied to: the stability of computational communities [259]; flow dominance in layout problems [215]; and hierarchical decomposition of systems [72].

8.41 Sober's Minimum Extra Information

In the field of the “simplicity” of scientific theories, Sober [418] rejected the idea of an absolute measure in favour of that of an ordering based on how much extra information would be needed to select an answer to a particular question: this is implicitly as relativised informativeness. Thus simplicity was to be relative to a question (represented by a set of possible answers). The theory that needs the least minimum extra information to select an answer to the question is deemed the simpler one. When one is judging theories with respect to a number of questions one must decide a weighting of the relative importance of the questions, to decide the overall simplicity.

Sober applies this to mathematical and logical fields by examining how the fundamental axioms are chosen. This is done by seeing how much information they contribute to the question of whether the axiom is true in our world or not. According to Sober

> *"This mirrors our belief that a contraction in the axiom set is a gain in simplicity. Moreover, a proof that the axioms are mutually independent is a proof that the axiom set is maximally simple; no axiom is redundant. And a proof that the axiom set is complete simplifies our view of the area being axiomatised, for it assures us that relative to the axiom set, every truth is redundant."*

Next Sober considers some logical properties of (extra-logical) relations by considering the informativeness of them relative to the general question of whether two objects are related. Thus he arrives at similar conclusions to Goodman (e.g. a symmetrical relation is simpler that an anti-symmetrical one etc.).

8.42 Sophistication

Koppel [268] defines "sophistication" as a measure of the structure of a string. He says:

> *"The minimal description of a string consists of two parts. One part is a description of the string's structure, and the other part specifies the string from among the class of strings sharing that structure (Cover 1985). The sophistication of a string is the size of that part of the description which describes the strings structure. Thus, for example, the description of the structure of a random string is empty and thus, though its complexity is high, its sophistication is low."*

Formally for finite strings the c-sophistication of S, a string, is $min\{|P| \mid \exists D\,(P, D)$ is a description of S and $|P| + |D| \leq H(S) + c\}$, where (P,D) is a *description* of S, if P is a total, self-delimiting program that computes S from the data, D and where H(S) is the algorithmic complexity of S, the minimum possible |P|+|D| such that (P,D) is a description of S.

Thus by allowing the data to be longer than the minimum, the program might be shorter. The idea is that any random, incompressible part might come from the data, and the sophistication measures the minimal length of the program that computes the structured aspect of the string.

Grassberger's effective measure complexity [194] can be seen as an entropic (and hence computable) version of sophistication. The relation to similar measures (e.g. the algorithmic information of section 8.2 on page 136 and logical depth section 8.4 on page 138) are covered in [269].

8.43 Stochastic Complexity

Rissanen [379] finds the idea of "shortest code length" (like algorithmic complexity) attractive but difficult to apply when modelling physical processes. He estimates the minimum code length of data encoded with a probabilistic model, using Shannon's coding theory.

This can be seen as a statistical and computable version of algorithmic information (section 8.2 on page 136) as well as an attempt to establish a principled trade-off between a model's complexity and error rate (see Appendix 6 - Complexity and Scientific Modelling).

Re-christened as the minimum description length (MDL) principle [378], it has been successfully applied to machine learning [377, 493].

8.44 Syntactic Depth

The deeper phrases are embedded in a statement (according to some syntax), the more difficult they are to understand. Identifying the ease of comprehension is one of the primary purposes of measures of syntactic complexity in formal language theory. In 1960 Yngve [483] proposed *depth of postponed symbols* as a measure of syntactic complexity, this was criticised by Miller and Chomsky [326] on formal grounds. They preferred the *degree of self-embedding* because it was "*… precisely the property that distinguishes context-free languages from the regular languages.*" Other measures proposed in [383] include depth and nesting.

The depth of a syntactic expression is the maximum number of arcs from root to leaf when represented in a tree form. This has nothing to do with either "logical depth" (section 8.4 on page 138) or "thermodynamic depth" (section 8.46 on page 161).

From the point of view of a modeller, depth is a useful way to stratify a space of expressions in a recursive language, as typically the number of possible expressions goes up exponentially with the depth. Although as [349] points out this could be done in any number of ways. Thus depth is relevant to the problem of induction whether by humans (see section 6.5 on page 129 and Appendix 6 - Complexity and Scientific Modelling) or in machine learning [111, 148, 333].

Syntactic depth as an indication of complexity has also been applied to menu design [239]; the difficulty of resolution of ambiguity [174] and circuit design [355, 356].

8.45 Tabular Complexity

Tabular complexity is an adaptation of Kolmogorov's ε-entropy [267] by Vitushkin [453]. It is a measure of the complexity of finite-state automata (see also section 8.32 on page 153). To calculate it one takes the tables representing the change of state and the output of the semi-group of the states of the automata and then decomposes these tables into smaller sub-tables, also allowing for the decomposition of the “wiring” (the connections) between these sub-tables etc. The minimum total volume obtainable is the tabular complexity, i.e. it is the volume of the most compact tabular representation.

Thus tabular complexity is similar to It is only applicable to processes modellable by finite automata (a proper subset of those computable by a Turing Machine). The tabular complexity can be very difficult to calculate but estimates can be produced by exhibiting specific tables.

8.46 Thermodynamic Depth

Seth Lloyd [291, 292] defines thermodynamic depth as $-\log q(\bar{\alpha})$, where $q(\bar{\alpha})$ is the long-term probability of the trajectory $\bar{\alpha} = \alpha_1\alpha_2\ldots\alpha_n$ (being a sequence of discrete states) arising by chance. This is intended as the total amount of (Shannon) information (section 8.36 on page 156) required to specify that trajectory.

This is closely related to the breadth of a system, which is defined as

$$(8) \quad -K\sum_{\bar{\alpha}} p(\bar{\alpha}) \log \frac{q(\bar{\alpha})}{p(\bar{\alpha})},$$

where $\bar{\alpha} = \alpha_1\alpha_2\ldots\alpha_n$ ranges over the possible trajectories of the system, p is a function of the time-specific probabilities of each trajectory and q is the long-term (equilibrium) probability of the trajectory arising from chance and K is a constant.

This is the unique form of a measure, *f*, with the following properties:

1. *f* is a function of *p* and *q*,

2. *f* is continuous in *p* and *q*,

3. *f* is additive along its trajectories in a similar way to Shannon entropy, i.e. if $\bar{\alpha} = \alpha_1 \alpha_2 \ldots \alpha_m$ and $\bar{\beta} = \beta_{m+1} \beta_{m+2} \ldots \beta_n$ are trajectories then

(9) $$f(\{p(\bar{\alpha}\bar{\beta}), q(\bar{\alpha}\bar{\beta})\}) = f(\{p(\bar{\alpha}), q(\bar{\alpha})\}) + \sum_{\bar{\alpha}} p(\bar{\alpha}) f\left(\{p(\bar{\beta}|\bar{\alpha}), q(\bar{\beta}|\bar{\alpha})\}\right)$$

4. $p(\bar{\alpha}) = q(\bar{\alpha}) \Rightarrow f(\{p(\bar{\alpha}), q(\bar{\alpha})\}) = 0$ i.e. discount information obtained from equilibrium.

When a trajectory has probability of 1, then the depth is the same as the breadth. This is further developed in [169] into a measure which combines algorithmic and entropic information.

8.47 Time and Space Computational Complexity

Computational complexity is now a much studied area with many formal results. It is usually cast as the order of the rate of growth of the resources needed to compute something compared to the size of its input.

Such time and space complexity measures are the most studied computational measures. Articles which include the word complexity often refer to these. They reflect the degree of effort required to compute a problem, independent of particular instances of that problem. They are fairly rough measure because they only give the degree of increase to within a constant factor, e.g. the order of the polynomial with which they increase. This is because of possible variations in the abstract computer that does the calculation.

Several variations of this have been proposed, including: extension to other fields like the real numbers [66]; continuous complexity models [318]; information based complexity (which adds a cost function to the information used) [440, 441, 442]; and using uniform rather than logarithmic size [211].

Applications include: social choice theory [251]; grammatical inference [155];learning [184]; simplification in logic [315]; feasibility of reasoning by a limited agent [281]; communication [235]; induction [111]; simulation [335]; control theory [486, 487]; perceptrons [500]; improving performance on 3-SAT problems [230]; and propagation in boolean circuits [425].

Summaries of the field can be found in [40, 41, 70].

8.48 Variety

A complex system is likely to exhibit a greater variety in terms of its behaviour and properties. Thus variety is an indication of complexity (though not always as sometimes a very complex system is necessary in order to maintain equilibrium). Variety can be measured by the simple counting of types, the spread of numerical values or the simple presence of sudden changes. In this way it overlaps with information (section 8.15 on page 144) and entropic (section 8.12 on page 143) measures.

Applications include: punctuated behaviour [38]; stability of ecosystems [353]; competing behaviours and control [357]; tree structures [237]; number of inequivalent models [89]; the interaction of connectivity and complexity [218]; and evolution [316].