

# **Syntactic Measures of Complexity**

**A thesis submitted to the University of Manchester for the degree of  
Doctor of Philosophy in the Faculty of Arts**

**1999**

**Bruce Edmonds**

**Department of Philosophy**

## **Table of Contents**

**Table of Contents - page 2**

**List of Figures - page 11**

**List of Tables - page 13**

**Abstract - page 14**

**Declaration - page 15**

**Notes of copyright and the ownership of intellectual property rights -  
page 15**

**The Author - page 16**

**Acknowledgements - page 16**

**1 - Introduction - page 17**

1.1 - Background - page 17

1.2 - The Style of Approach - page 18

1.3 - Motivation - page 19

1.4 - Style of Presentation - page 20

1.5 - Outline of the Thesis - page 21

**2 - Models and Modelling - page 23**

2.1 - Some Types of Models - page 25

2.2 - Combinations of Models - page 28

2.3 - Parts of the Modelling Apparatus - page 33

2.4 - Models in Machine Learning - page 38

2.5 - The Philosophical Background to the Rest of this Thesis - page 41

### **3 - Problems and Properties - page 44**

- 3.1 - Examples of Common Usage - page 44
  - 3.1.1 - A case of nails - page 44
  - 3.1.2 - Writing a thesis - page 44
  - 3.1.3 - Mathematics - page 44
  - 3.1.4 - A gas - page 44
  - 3.1.5 - An ant hill - page 45
  - 3.1.6 - A car engine - page 45
  - 3.1.7 - A cell as part of an organism - page 46
  - 3.1.8 - Computer programming - page 46
- 3.2 - Complexity as a Comparison - page 46
  - 3.2.1 - The emergence of life - page 47
- 3.3 - What the Property of Complexity Could Usefully Refer to - page 47
  - 3.3.1 - Natural systems - page 47
  - 3.3.2 - The interaction of an observer with a system - page 53
  - 3.3.3 - Patterns - page 55
  - 3.3.4 - The modelling relation - page 56
  - 3.3.5 - A model with respect to a specified framework - page 56
- 3.4 - Some Unsatisfactory Accounts of Complexity - page 57
  - 3.4.1 - Size - page 57
  - 3.4.2 - Size of rules - page 58
  - 3.4.3 - Minimal size - page 58
  - 3.4.4 - Processing time - page 59
  - 3.4.5 - Ignorance - page 60
  - 3.4.6 - Variety - page 61
  - 3.4.7 - Midpoint between order and disorder - page 62
  - 3.4.8 - Improbability - page 63
  - 3.4.9 - Expressivity - page 65
  - 3.4.10 - Dimension - page 65
  - 3.4.11 - Ability to surprise - page 66
  - 3.4.12 - Logical strength - page 66

3.4.13 - Irreducibility - page 67

3.5 - Complexity is Relative to the Frame of Reference - page 68

3.5.1 - The level of application - page 68

3.5.2 - Goals - type of difficulty - page 69

3.5.3 - Atomic parts - page 69

3.5.4 - The language of description - page 69

## **4 - A Definition of Complexity - page 72**

4.1 - Aspects of the Definition - page 75

4.1.1 - Identity of a system - page 75

4.1.2 - Atomic components - page 76

4.1.3 - Difficulty - page 77

4.1.4 - Formulating overall behaviour - page 78

4.1.5 - Complexity vs. ignorance - page 79

4.1.6 - As a gap between the global and local - page 80

4.1.7 - The comparative nature of complexity - page 80

4.1.8 - The existence of complexity - page 81

4.1.9 - Relativisation to a language - page 81

4.2 - Examples - page 81

4.2.1 - The flight behaviour of a herd - page 82

4.2.2 - Cellular automata - page 82

4.3 - Relationship to Some Other Formulations - page 83

4.3.1 - Number of inequivalent descriptions - page 83

4.3.2 - Effective measure complexity - page 84

4.3.3 - Computational complexity - page 84

4.3.4 - Algorithmic information complexity - page 84

4.3.5 - Shannon entropy - page 85

4.3.6 - Crutchfield's "topological complexity" - page 85

## **5 - Applications of Complexity to Formal Languages - page 86**

5.1 - Types of Complexity Involving Formal Languages - page 86

5.2 - Expected Properties of "Analytic Complexity" - page 87

- 5.2.1 - Independent of the particular symbols used - page 87
- 5.2.2 - The complexity of sub-expressions should be less than the whole - page 87
- 5.2.3 - Expressions with no repetitions are simple - page 88
- 5.2.4 - Small size should limit the possible complexity - page 89
- 5.2.5 - There should be no upper limit to complexity if the language is suitably generative - page 89
- 5.2.6 - The complexity of irrelevant substitutions - page 90
- 5.2.7 - The complexity of relevant relating of expressions - page 91
- 5.2.8 - Decomposability of expressions - page 91
- 5.3 - Measures of Analytic Complexity - page 93
  - 5.3.1 - Notation - page 93
  - 5.3.2 - Weak complexity measures - page 95
  - 5.3.3 - Weak complexity measures where simple repetition does not increase complexity - page 100
  - 5.3.4 - Weak complexity measures that respect the subformula relation and where simple repetition does not increase complexity - page 102
  - 5.3.5 - Strong complexity measures - page 104
- 5.4 - The Cyclomatic Number as a Measure of Analytic Complexity - page 106
- 5.5 - Layers of Syntax and Complexity - page 108
  - 5.5.1 - Example 1 - a supply of variable names - page 109
  - 5.5.2 - Example 2 - WFFs of the implication fragment - page 110
  - 5.5.3 - Example 3 - The implicational fragment of E - page 111
  - 5.5.4 - Discussion of syntactic structures - page 112
- 5.6 - Application to Axioms and Proof Systems - page 113
  - 5.6.1 - Axiom complexity - page 113
  - 5.6.2 - Proof complexity - page 117
- 5.7 - Application to Simplification - page 120
  - 5.7.1 - Searching over equivalent expressions within a language - page 120
  - 5.7.2 - Searching over equivalent derivations within a language - page 121
  - 5.7.3 - Specialising the syntactic level - page 121

5.7.4 - Searching over equivalent languages - page 123

5.7.5 - Simplification via trade-offs with specificity and accuracy - page 124

## **6 - Philosophical Applications - page 126**

6.1 - Complexity and Relevance - page 126

6.2 - Complexity and Emergence - page 126

6.3 - Complexity and Language - page 128

6.4 - Complexity and Representation - page 128

6.5 - Complexity and “Simplicity” - page 129

6.6 - Complexity and Evolution - page 130

6.7 - Complexity and Holism - page 131

6.8 - Complexity and System Identity - page 132

6.9 - Complexity and Society - page 133

## **7 - Conclusion - page 134**

7.1 - Further Work - page 134

## **8 - Appendix 1 - A Brief Overview of Some Existing Formulations of Complexity - page 136**

8.1 - Abstract Computational Complexity - page 136

8.2 - Algorithmic Information Complexity - page 136

8.3 - Arithmetic Complexity - page 138

8.4 - Bennett's ‘Logical Depth’ - page 138

8.5 - Cognitive Complexity - page 139

8.6 - Connectivity - page 140

8.7 - Cyclomatic Number - page 140

8.8 - Descriptive/Interpretative Complexity - page 141

8.9 - Dimension of Attractor - page 141

8.10 - Ease of Decomposition - page 142

8.11 - Economic Complexity - page 142

## Syntactic Measures of Complexity

- 8.12 - Entropy - page 143
- 8.13 - Goodman's Complexity - page 143
- 8.14 - Horn Complexity - page 143
- 8.15 - Information - page 144
- 8.16 - Information Gain in Hierarchically Approximation and Scaling - page 145
- 8.17 - Irreducibility - page 145
- 8.18 - Kemeny's Complexity - page 146
- 8.19 - Length of Proof - page 146
- 8.20 - Logical Complexity/Arithmetic Hierarchy - page 146
- 8.21 - Loop Complexity - page 147
- 8.22 - Low Probability - page 148
- 8.23 - Minimum Number of Sub Groups - page 148
- 8.24 - Minimum Size - page 149
- 8.25 - Mutual Information - page 151
- 8.26 - Network Complexity - page 151
- 8.27 - Number of Axioms - page 152
- 8.28 - Number of Dimensions - page 152
- 8.29 - Number of Inequivalent Descriptions - page 152
- 8.30 - Number of Internal Relations - page 153
- 8.31 - Number of Spanning Trees - page 153
- 8.32 - Number of States in a Finite Automata - page 153
- 8.33 - Number of Symbols - page 154
- 8.34 - Number of Variables - page 155
- 8.35 - Organised/Disorganised Complexity - page 155
- 8.36 - Shannon Information - page 156
- 8.37 - Simplicity - page 156
- 8.38 - Size - page 157

- 8.39 - Size of Grammar - page 157
- 8.40 - Size of matrix - page 158
- 8.41 - Sober's Minimum Extra Information - page 158
- 8.42 - Sophistication - page 159
- 8.43 - Stochastic Complexity - page 160
- 8.44 - Syntactic Depth - page 160
- 8.45 - Tabular Complexity - page 161
- 8.46 - Thermodynamic Depth - page 161
- 8.47 - Time and Space Computational Complexity - page 162
- 8.48 - Variety - page 163

## **9 - Appendix 2 - Longer Proofs - page 164**

- 9.1 - (Non-existence of) Complexity Measures on Strings - page 164
- 9.2 - Cyclomatic Number as a Lower Bound for Minimal Damage Cut - page 169
- 9.3 - Decomposition of Formulas into Complexes - page 169
- 9.4 - Generating a Measure from a Function on the Complexes - page 170
- 9.5 - Three Conditions that are Equivalent on a Weak Complexity Measure - page 175

## **10 - Appendix 3 - Formalisation of Syntactic Structure - page 182**

- 10.1 - Formalisation - page 182
  - 10.1.1 - The syntax of trees - page 182
  - 10.1.2 - The syntax of rules - page 182
  - 10.1.3 - The syntax of syntactic structures: - page 182
  - 10.1.4 - Generation from syntactic structures - page 183
  - 10.1.5 - Production from trees - page 183
  - 10.1.6 - Complete production - page 184
  - 10.1.7 - Complete productive generation from syntactic structures - page 184
- 10.2 - The Expressivity of Syntactic Structures - page 184
- 10.3 - Flattening Syntactic Structures - page 187

**11 - Appendix 4 - A tool for exploring syntactic structures, complexity and simplification - page 188**

11.1 - Overview - page 188

11.2 - Examples - page 188

11.3 - Programming - page 190

11.4 - Interface - page 192

**12 - Appendix 5 - A comparison of different rankings of logical formula - page 193**

**13 - Appendix 6 - Complexity and Scientific Modelling - page 199**

Overview - page 199

Complexity - page 200

A Framework for Analysing Modelling - page 201

Other Formulations of Complexity - page 202

Order and Disorder - page 203

Noise - page 206

Complexity vs. Information - page 207

Complexity and Induction - page 207

Conclusion - page 208

**14 - Appendix 7 - Complexity and Economics - page 210**

What is Complexity? - page 210

“Complexity” in economics - page 210

The “Sciences of Complexity” - page 210

Complexity *per se* - page 211

The effects of complexity on modelling by agents - page 213

Ideal rationality and perfect information - page 213

Ideal rationality and noisy information - page 213

Ideal rationality and inadequate information - page 214

Bounded rationality and inadequate information - page 214

The effects of modelling by agents on complexity - page 215

## Syntactic Measures of Complexity

Ignoring the process of modelling by economic agents - page 215

Including the process of modelling by economic agents - page 215

Towards dealing with the complexity of modelling agents - modelling modelling -  
page 217

The Form - meaning distinction - page 217

The complexity, specificity, error trade-off - page 218

The modelling language - page 219

Processes of model development - page 219

Some future directions for economic modelling - page 220

Applying our model of modelling to ourselves - page 220

Relatively new (non-numerical) techniques - page 221

Conclusion - complexity again - page 222

## **15 - References - page 223**

## List of Figures

- Figure 1: Entity,  $W$ , using  $A$  as a model of  $B$  - page 24
- Figure 2: An illustration of the syntactic view of models - page 28
- Figure 3: A semantic picture of modelling (from Giere in [176]) - page 29
- Figure 4: The picture with explanatory and phenomenological models - page 30
- Figure 5: The picture with the data model added - page 31
- Figure 6: The models used in chemical simulation (adapted from [201]) - page 32
- Figure 7: Model specification and semantics - page 34
- Figure 8: Relating model-structures via the ‘subsumption’ of model contents - page 36
- Figure 9: Relating model-structures via approximation of model contents - page 36
- Figure 10: Relating model-structures via ‘chaining’ - page 37
- Figure 11: The ‘modelling relation’ from [385] - page 41
- Figure 12: Observer–system pair - page 53
- Figure 13: An observer-system pair with multiple models - page 54
- Figure 14: Complete order, chaos and complete disorder - page 62
- Figure 15: Presumed graph of disorder against complexity - page 62
- Figure 16: Possible diagrammatic inclusions - page 63
- Figure 17: Finding a description to cover a data model - page 73
- Figure 18: Measuring the difficulty of finding a model description - page 74
- Figure 19: A formula represented as a tree - page 107
- Figure 20: A formula represented as a collapsed tree - page 108
- Figure 21: A single layer syntactic structure - page 110
- Figure 22: A two layer syntactic structure - page 111
- Figure 23: The actual structure of the assertion axiom - page 111
- Figure 24: A three layer structure - page 112
- Figure 25: The proof of identity in Lukasiewicz’s single axiom system - page 117
- Figure 26: An example proof net - page 118
- Figure 27: A setup for varying languages - page 124
- Figure 28: The arithmetic complexity hierarchy - page 147
- Figure 29: A window on a single syntactic structure - page 189

## Syntactic Measures of Complexity

Figure 30: Three inter-dependent syntactic structures in separate windows - page 190

Figure 31: The hierarchy of generator classes - page 191

Figure 32: Three patterns - after [194] - page 204

Figure 33: The same patterns with some suggested inclusions - page 205

Figure 34: A diagram of the relationship between relevant types of complexity - page 217

Figure 35: Form and meaning distinction - page 218

Figure 36: Using a genetic population to model an economic agent - page 220

## List of Tables

- Table 1: Some properties ascribed to the model syntax and semantics - page 38
- Table 2: Formulas up to size 6 sorted by size - page 193
- Table 3: Formulas up to size 6 sorted by number of distinct variables - page 194
- Table 4: Formulas up to size 6 sorted by maximum depth - page 195
- Table 5: Formulas up to size 6 sorted by breadth (number of variables) - page 196
- Table 6: Formulas up to size 6 sorted by cyclomatic number of their minimal graph  
- page 197
- Table 7: Rankings of formula by different methods (bold lines group equally  
ranked formulas) - page 198

## **Abstract**

This thesis analyses the conception and measurement of complexity and then applies it to some aspects of formal languages.

It starts with a review of the philosophy of modelling. It continues by considering some simple examples to establish intuitions about the common use of ‘complexity’ and goes on to examine what complexity can usefully be attributed to as a property. It argues that it most useful as an attribute of the specification of a model. Some unsatisfactory accounts of complexity are discussed as motivation for the definition of complexity that is then suggested. Some other accounts of complexity are shown to be special cases of the one suggested here.

This approach is then applied to formal languages. A set of properties of analytic complexity are set-out. The set of measures which satisfy these properties is formally investigated. The cyclomatic number of a representation of expressions is put forward to model analytic complexity. In order to analyse shifts in complexity a formal device called syntactic structures is defined. This consists of layers of syntaxes, each with its own production rules which generate the contents of that layer. Each syntactic structure can use substitutions from lower such structures, so that collections of such structures can form hierarchies.

These approaches to are then applied to axiomatic and proof theoretic aspects of logic. Some potential methods of simplification are suggested. Finally some remarks are made about the philosophical applications of this approach.

The appendices include a survey of measures of complexity in the literature; a brief description of a software tool written to explore syntactic structures, two relevant papers on the application of these ideas to scientific modelling and economics, and an extensive bibliography.

## **Declaration**

**No portion of the work referred to in this thesis has been submitted in support of an application for another degree or qualification of this or any other university or other institution of learning.**

## **Notes of copyright and the ownership of intellectual property rights**

- (1) Copyright in the text of this thesis rests with the Author. Copies (by any process) either in full, or of extracts, may be made **only** in accordance with instructions given by the Author and lodged with the John Rylands University Library of Manchester. Details may be obtained from the Librarian. This page must form part of any such copy made. Further copies (by any process) of copies made in accordance with such instructions may not be made without permission (in writing) of the Author.
- (2) The ownership of any intellectual property rights which may be described in this thesis is vested in the University of Manchester, subject to any prior agreement to the contrary, and may not be made available for use by third parties without the written permission of the University, which will prescribe the terms and conditions of any such agreement.

Further information on the conditions under which disclosures and exploitation may take place is available from the Head of Department of the department of Philosophy.

## **The Author**

The author gained his first degree in mathematics at Oxford in 1983. He then worked for an educational charity and taught abroad until 1991. He then started studying for this doctorate part-time. In 1994 he got his present job as Senior Research Fellow in Logic and Formal Methods at a research unit of the Manchester Metropolitan University called the Centre for Policy Modelling, which he continues to hold. His work there includes research into the methodology and philosophy of simulation, the logic of the declarative programming language they use (SDML), evolutionary techniques of modelling cognitive agents and the philosophy of science. His publications and the bibliography on complexity that he maintains may be found at URL: <http://www.cpm.mmu.ac.uk/~bruce>. He is also an editor of the newly launched academic web journal, the *Journal of Memetics – Evolutionary Models of Information Transmission*.

## **Acknowledgements**

I would like to acknowledge my supervisor, John Chidgey for stimulating discussions and some tolerance and my colleagues at the Centre for Policy Modelling: Scott Moss, Steve Wallis, and Helen Gaylard for many discussions on almost every topic under the sun except complexity.

For Kim, Patsy, Ruth and Orlanda.

# 1 Introduction

## 1.1 Background

The use of “complexity” as a label for an object of study is fairly recent. Before the middle of this century it appeared to be merely an antonym of “simplicity” (where “simplicity” here means *that property that guides the rational choice between competing theories that are equally supported by evidence* - see section 6.5 on page 129). This sort of “simplicity” follows in a philosophical tradition that is traced back to William of Occam’s famous razor<sup>1</sup>. Although there has been some implicit interest with complexity amongst logicians since the 1930s<sup>2</sup> associated with limiting *size*, the first direct discussion of “complexity” that I have found is in Weaver’s paper in 1948: “Science and Complexity” [464].

In the last 10 years there has been a tremendous amount of interest in “complex systems” of various kinds<sup>3</sup> and consequently there has been some interest in “complexity” *per se*. Unfortunately<sup>4</sup>, this has caused an overloading of the term so that frequently it seems only to tag a work as inhabiting an intellectually desirable area<sup>5</sup>. Thus “complexity” appears in the titles of books and articles which, on inspection, are only tenuously related to it. In many of these works it is sufficient for them that they are considering situations where previous analytical techniques fail<sup>6</sup> or merely that some process of elaboration is involved.

If an author talks about a *complexity measure* then usually more justification is made for the term's use. Many of these measures, however, have a post-hoc flavour; a device is invented or required that has something vaguely to do with complexity and, for the want of a better term, is given that label. Some other, more serious attempts, seem skewed by a prior formal requirement.

---

1. “Entities are not to be multiplied beyond necessity” – previously invoked by Durand de Saint-Pourcain.

2. See section 5.6.1 on page 113.

3. For an overview of this see [168].

4. Unfortunately for its analytic use that is, the transfer of such terms to the public domain has other social uses [455].

5. This is the essence of Horgan’s critique in [233].

6. Many publications that can be grouped under the “Sciences of Complexity” banner fall into this category.

Some serious attempts at tackling the subject have been made, notably in the mid '70s in general systems theory and recently in the study of chaotic processes. The former were more generally applicable but seemed to peter out due to a lack of consensus and the limited number of practical results gained. The later is still in its infancy but is naturally concerned with the specific problems of modelling chaotic physical processes.

References to complexity are now spread over a wide range of fields, including: biology, mathematics, physics, computation, logic, economics, software design, philosophy, general systems, management science, psychology and linguistics, although the two fields of chaotic process and computational complexity account for many of them.

## 1.2 The Style of Approach

*Firstly*, I will focus my discussion on what complexity *is* rather than what might *cause* it. It would seem vain to attempt an investigation of such causes *in general*; if there were but a few general causes then we might well have made more direct progress in tackling the problem of complexity where it arises. Having said that, making progress towards pinning the concept of complexity down would be a first step towards a more coherent and cumulative study into its causes in particular contexts<sup>7</sup>.

*Secondly*, within this framework I will be approaching the subject from a pragmatic perspective. What I will be looking for are verbal and formal models of complexity that are generally applicable and useful. A useful model is, trivially, one which helps you achieve your goals. In this case it will imply models of complexity which are clear, manipulable (both formally and informally), applicable in a wide range of circumstances and reflect the motivations listed below. I will not go much further into the meaning of utility, as this would go beyond the scope of this thesis. This pragmatic perspective must be borne in mind when interpreting the discussion on the attribution of complexity in section 3.3 on page 47 and its definition in section 4 on page 72.

Such a pragmatic modelling approach seems to fall somewhat between prescription and description. A useful model may be considered prescriptive if you subscribe to the same goals and framework as that of the model. Further than that I do not wish to prescribe usage of “complexity”, except to point out some of the less helpful consequences of its unthoughtful usage. It may be considered as descriptive because it can

---

7. For an example of such a call for the formal study of complexity see Casti in [89].

be used as a point of reference to relate different examples in an illuminating way. It thus may be used as a descriptive *framework*.

*Thirdly*, I will apply these models. Although I will consider several examples and consider the philosophical applications, the main target of this study is the application of such models to formal syntactic systems, since this is where I see the numeric quantification of complexity providing the most insight.

### 1.3 Motivation

The motivation for this study is as follows:

1. *To lay bare assumptions and relativisations involved in the usage of the term.*

As will become clear, the term is frequently used in a vague way, making many assumptions concerning the aims, language and foundation of the subject matter being discussed. These assumptions are sometimes not justified. An example where the vagueness of the term can frustrate progress is in the study of evolution where the issues of whether complexity increases with evolution and how this may happen are debated with cross purposes (see section 6.6 on page 130). I have only found one instance where it was suggested that the rigorous definition of complexity might be counter-productive [260].

2. *To allow a meaningful comparison between different formulations of complexity across different fields of study.*

At the moment, apart from systems theory, models of complexity tend to be formulated with quite specific purposes in mind<sup>8</sup>. The result of this is that there are only vague and unformalised connections between such models developed in different fields. This must impede progress and mean that there is needless repetition. A common explanatory framework would make clear the similarities and differences that exist between such formulations across subject boundaries.

---

8. This is particularly clear in physics, in the development of models of complexity applicable to chaotic processes, e.g. [457]

3. *To lay the foundations for formalisations of complexity, in different circumstances.*

A formalisation is often more useful for the development of ideas if its components have a clear meaning. These philosophical investigations can provide some of this framework.

4. *To aid the formulation of deeper insights into possible causes of complexity.*

Once one starts to develop models within a coherent framework, it is often easier to build upon previous results and relate informal discussions on the subject given a concrete referential framework that such a model would provide.

5. *To allow the development of systematic approaches to simplification.*

Simplification is the pot of gold at the end of the complexity rainbow. Systematic approaches to simplification (both human and automatic), would be of immense use in academic study, almost irrespective of how marginally it was. Such techniques would be applicable to both the object and method of study, as well as highlighting the cases where simplification is not possible.

This thesis aims to make some progress towards these goals.

## **1.4 Style of Presentation**

Given the above motivation, and that my approach to the problem of defining, formalising and using the concept of complexity will be pragmatic in nature, I am concerned that this thesis should not be only a survey of existing models of complexity<sup>9</sup>. So I will relegate a survey of such papers to Appendix 1, which will summarise the main approaches and give the relevant references. Thus, frequently, when discussing these approaches, I will not swamp the main text with citations but refer to the appropriate section of this appendix, where a more systematic account of each idea can be given and the full references given. In this way I hope to improve the readability of the main text, while retaining its scope.

I have also relegated a lot of the proof details, formalisation, and two papers summarising some of philosophical applications of this approach to complexity to the appendices. Thus in this thesis the appendices contain much of the content of this thesis,

---

9. If existing measures were more established and demonstrably useful my approach would be different.

but I felt that this arrangement made the central thrust of the thesis more focused and accessible.

## **1.5 Outline of the Thesis**

Section 2 reviews some of the thought on models and modelling in the philosophy of science and the machine learning communities. It then analyses some of the components of the modelling apparatus on a model-theoretic basis, and establishes some terminology that will be used in this thesis. Then it uses this analysis to categorize three ways that different models are related. Finally it briefly state the philosophical position from which the thesis is written.

Then section 3 on page 44 starts with some examples in order to focus on the properties of complexity. It argues that complexity is only usefully attributable to model descriptions relative to a language of expression. It also considers and argues against a number of other basic approaches.

In section 4 on page 72, I present my approach, discuss it and the terms I mention. I also give some examples and relate this approach to some other approaches.

In section 5 on page 86, I apply this approach to formal languages. Here I focus upon what one might mean by “analytic complexity” and what properties one might expect of it. I then present a way of structuring formal languages to enable the analysis of complexity in such languages, which I then apply to axiom complexity and proof complexity. I finish this section by considering some approaches to systematic simplification.

In the last section (section 6 on page 126), I consider some philosophical applications of my approach and conclude in section 7 on page 134.

There are several appendices: section 8 on page 136 is an overview of the main approaches to the formulation of complexity in different fields, section 9 on page 164 holds the details of some of the longer proofs; section 10 on page 182 gives the detailed formalisation for layered syntactic structures; section 11 on page 188 describes a computer-base tool I wrote to help explore the properties of syntactic structures; section 12 on page 193 lists tables with rankings of the complexity of logical formulas according to different measures; section 13 on page 199 is a paper on the philosophical application of complexity to scientific modelling, section 14 on page 210 is an unravelling of various

## Syntactic Measures of Complexity

conceptions of complexity in economics and section 15 on page 223 comprises the references.

## 2 Models and Modelling

The term ‘model’ is used in many different situations and in many different ways. This causes some difficulties when trying to discuss the uses of models – what Wartofsky calls the “Model Muddle”:

*“The symptom of the (model) muddle is the proliferation of strange and unrelated entities which come to be called models.” [462] p.1*

He goes on to identify the root cause of this proliferation, the fact that “*anything can be a model of anything*” [462] p.4 provided only that:

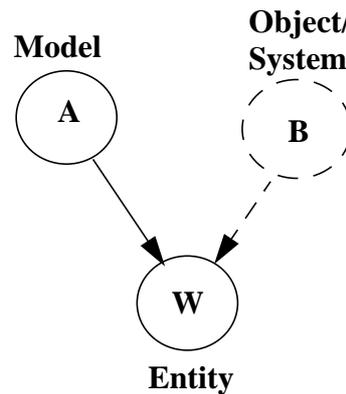
*“... every case of being taken as a model involves a restriction with respect to relevant properties.” [462] p.6.*

Wartofsky does not mean that anything can be used as a model for anything for any purpose (for example, a robin used as a model of urban development), but that one can not rule out  $A$  being used as a model for  $B$  just by the nature of  $A$  and  $B$  alone. There may be some properties of  $A$  and  $B$  which when selected as relevant (for some purpose) will legitimise  $A$  being taken as a model of  $B$ . This is implicit in his statement that

*“The limits on modelling are therefore the limits of our conceptions of what respects are relevant to what purposes.” [462] p.6*

Of course, these conceptual limitations might be quite strong! We might be limited in our conceptions by our experience of the world and our reasoning about it, for example. However, our conceptions of the world and hence which things are relevant to which might be very strange – it is conceivable that for any objects or systems,  $A$  and  $B$ , *someone* might be have a (possibly mistaken) conception of the relevance of their properties so that that person might use  $A$  as a model of  $B$  (despite the fact that the rest of us might see this as pointless).

Thus modelling can be seen as triadic relation,  $M(S, A, B)$  –  $S$  takes  $A$  as a model for  $B$ , by finding or choosing relevant properties of both,  $R(A)$  and  $R(B)$  such that  $R(A) \subseteq R(B)$ . This is illustrated in figure 1. To avoid repetition I will refer to the object or system being modelled as simply the *object*, unless this is unclear.



**Figure 1.** Entity, W, using A as a model of B

Apostel relativises this relation further, by making explicit the purpose (or ‘modelling goal’) of the person in using a model. Thus he writes the basic relation as  $M(S, P, A, B)$ , meaning that “*The subject S takes, in view of the purpose P, the entity A as a model for the prototype B.*” [16] p. 4.

These are, however, very general definitions which do not tell us much about the nature of models as actually used by people; how to distinguish between useful and less useful models, or how they should use them. To do this one has to narrow the context down and subsume some of the necessary or accepted limitations on our conceptions of relevance in the places where models are used in a deliberate manner. The area where this has been most studied is in the practice and theory of science.

Models in the philosophy of science are usually distinguished from theories. However, the distinction is not always clear cut and some authors slip from one to the other. Three examples of philosophers who use the terms differently are: Apostel [16] who allows a theory to be a model, Hesse [216] who argues that a theory is meaningless if it does not *include* a model and van Frassen who says “*To present a theory is to specify a family of structures, its models...*” [161] p. 64). On the whole the use of the term ‘model’ implies something relatively concrete, possibly visualisable, approximate, specific and tractable whereas a ‘theory’ implies a more general, abstract and reliable construction. In the social sciences it is more common to use the term ‘model’ for things that would be called theories in the physical sciences (according to Braithwaite, 1962, who deplores the practice).

The distinction between models and theories is clearest in those who take a ‘model-theoretic’ approach. Here a strong analogy is made with the Tarskian logical semantics – the theory is likened to a logical theory describable in terms of a set of axioms along with an inference procedure and a scientific model is likened to a logical model, that is a formal structure which satisfies the theory. According to DaCosta and French in [129] this approach was first introduced by Beth [61, 62, 63] and Suppes [430, 432]. Others wish to deliberately conflate these entities in the search for a more basic structure, from which the ‘headline’ entities might be built (e.g. Giere [176] and Wartofsky [462]). My approach will tend towards the later, since I am not concerned here with the truth of such entities but with the complexity that arises as a result of the process of modelling. However, I will maintain a clean distinction between the syntactic and semantic parts of the modelling apparatus.

## 2.1 Some Types of Models

I will briefly outline some of the types of models that have been identified in the philosophy of science literature, before going on to discuss some of the relations between them in the next subsection (Section 2.2).

Firstly, models may be categorised according to the medium in which they are expressed. In this way we have physical models including scale models (the first type of model of Achinstein in [6] and the second sense of model of Suppe in [427]) and Irving Fisher’s model of the money supply using coloured liquids (as documented by Morgan in [328]); mathematical models<sup>10</sup> where the language of mathematics is the primary medium for defining the content of the model; computational models where the model is encoded as a computer program with the computation somehow modeling the object process; and linguistic models where normal (or pseudo-formal) spoken or written language is used to specify the model<sup>11</sup>.

Secondly models are distinguished according to whether the term model applies to the sentences and formal theories that describe the model or to the referent of those

---

10.Unfortunately ‘mathematical model’ has two connotations: that of being expressed in terms of mathematics and that of being formal in contrast to analogical models.

11.There are also, of course, logical models but, in order to prevent confusion I will only use this phrase to denote the sort of models found in metalogic.

sentences. The former have been called ‘syntactic models’ or (more precisely) the syntactic approach to models (e.g. Ayer [32] and Duhem [143]). The later ‘semantic models’ or the semantic approach to models is where the models are the entities described by (or that satisfy) the description – these can be further divided into those which envisage these semantics to be akin to the semantics in metalogic where a formal structure is a model when it satisfies the model specification (e.g. Sneed [417], Suppe [427]<sup>12</sup> and Suppes [433]), and those which include a wider, linguistic semantics (e.g. Giere [175]).

In a different way analogical and mathematical models are distinguished. An analogical (or iconic) model involves a (possibly incomplete) analogy between some properties of the model and the object or system being modelled or between the relationships between the properties of the model and the relationships between the properties of what is modelled (as Sellars prefers to define it in [406]). These include the iconic models of Hesse [216] and Suppe [427] as well as model<sub>2</sub> of Cushing in [126]. Morgan [328] further distinguishes between ‘ready made’ and ‘designed’ analogical models. An important aspect of the iconic models is claimed to be that they allow the theory to become predictive, thus Hesse says:

*“(an iconic model is) any system whether buildable, picturable, imaginable or none of these which has the characteristic of making a theory predictive.” [216] p.19*

A mathematical model implies a more formal structure, which may be an approximation or simplification of the underlying theory (including Cushing’s model<sub>1</sub> [126], the ‘theoretical model’ of Achinstein in [5] and Suppe’s first sense of model in [429]). Achinstein’s theoretical model is a type of mathematical model but implies a slightly greater existential commitment, it is somehow an exemplar or expression of a theory. Some authors allow whole series of model types differing in the ascribed existential commitment (e.g. Wartofsky [462]). Kuhn allows models a similar range of epistemic commitment:

*“Models, ... are what provide, the group with preferred analogies or, when deeply held, with an ontology.” [274] p.463*

---

12. Suppe confusingly calls these ‘mathematical models’.

Cartwright [84] distinguishes between ‘explanatory models’ which have a broad scope and are used to explain the cause(s) of events (but, she claims, are not descriptively true) and the phenomenological laws which are accurate representations of the phenomena but which do not provide satisfactory explanations. In this conception there is a hierarchy of models from the dirty-but-true up to the general-but-inapplicable:

*“To explain a phenomenon is to find a model that fits it into the basic framework of the theory and that allows us to derive analogues for the messy and complicated phenomenological laws which are true of it.” [84] p.152*

In many of these conceptions (especially those of the semantic approach) a model resides within the framework of a theory. Redhead [372] distinguishes two ways in which this can occur: by simplifying the theory to become more tractable (*impoverished*) or by filling in missing detail to *enrich* the theory.

There are what are called ‘Tinker Toy’ or ‘Floating’ Models (Post, 1975 as reported by Redhead in [372] and Cushing’s model<sub>3</sub> [126]). These are models which are neither verified against known data nor related to a deeper theoretical framework – they just ‘float’ somewhere in between. Despite these having been denigrated in terms of their uncertain status by Post, they have been recently defended by Redhead as a valuable tool to aid the testing of theories [372].

Finally there are: Suppes’ ‘data model’ [431], Kemeny’s “intended factually true description” [255] and the abstract models not intended to be true of an natural thing (Achinstein’s third type of model in [6]). In addition to the above, models could also be divided into the subject areas they originate from or are used in.

Even the above list is by no means comprehensive. Every time a philosopher characterises or defines a model in a different way one could give it a new label and add it to the confusion (Chao in [103] lists no fewer than 30 characterisations). What turns out to be more important than definitions is the different functions and uses that such ‘models’ are put to and the ways in which they are deployed (both singly and in combination) by practising scientists. As Apostel summarises it:

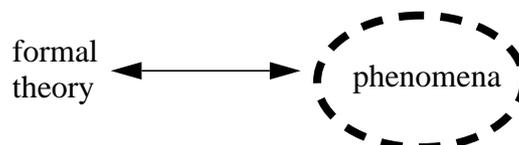
*“(a model is) ... any subject using a system A that is neither directly nor indirectly interacting with a system B, to obtain information about the system B, is using A as a model for B.” [16]*

## 2.2 Combinations of Models

The knowledge gained from applying analogical models is far from certain, unless you can be sure they are being applied to a situation that is essentially identical to a previous application. The logical empiricists at the beginning of the 20<sup>th</sup> century wished to put science onto a surer footing: analogies were denigrated as mere heuristics and the focus shifted towards the use of formal systems. As Braithwaite puts it:

*“Analogy can provide no more suggestions of how the theory might be extended; and the history of science tells us that while some analogical suggestions have led to valuable extensions of the theory, others have led to dead ends” [73] p.230*

The ideal form of scientific knowledge was a formal system of axioms whose logical consequences were to be related to the observed phenomena as directly as possible (figure 2). Thus one had a fairly simple picture with the formal theory being manipulated by syntactic manipulations in order to make predictions about phenomena. As a more descriptive approach to the workings of science has been adopted by many philosophers, there has been a progressive elaboration of this picture.

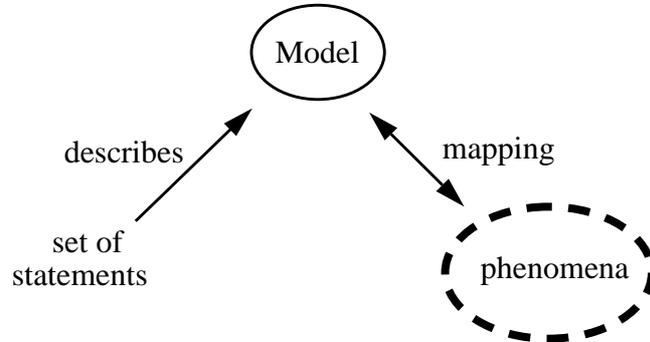


**Figure 2.** An illustration of the syntactic view of models

Tarskian Logical Semantics provided such an step. Scientific models can be identified with logical models and scientific theories with logical theories. The model, according to this account, is a structure which satisfies the statements that define the theory. This model is then related to the phenomena under study. Two corollaries of this are that different descriptions may identify the same model and that a theory may have several models. This picture of models as semantic entities is illustrated in figure 3 and summarised in the following quote from van Frassen:

*“... the language used to express the theory is neither basic nor unique; the same class of structures could well be described in radically different*

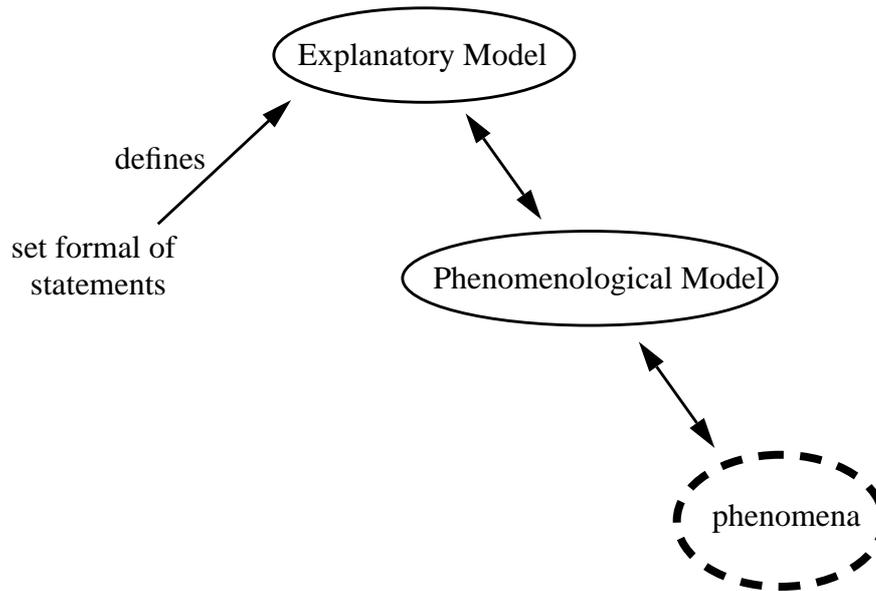
*ways, each with its own limitations. The models occupy centre stage.” [161]  
p.44*



**Figure 3.** A semantic picture of modelling (from Giere in [176])

Cartwright’s distinction between explanatory and phenomenological laws [84] can be seen as adding a further layer to this picture. The phenomenological laws relate more directly to the phenomena – they have the function of organising the phenomena but do not provide suitable material to construct explanations. Explanatory laws, on the other hand, provide explanations but are not descriptive of the phenomena – they are not literally *true*. This account is interpreted in terms of layers of models by Hughes in [236] and illustrated below in figure 4. In this way the explanatory laws acts as models of the phenomenological laws. Cartwright says:

*“To explain a phenomenon is to find a model that fits it into the basic framework of the theory and that allows us to derive analogues for the messy and complicated phenomenological laws which are true of it.” [84] p.152*



**Figure 4.** The picture with explanatory and phenomenological models

A further layer is added if one accepts the importance of the distinction between data and phenomena as discussed by Bogen and Woodward in [68]. They point out that data is used more for prediction and the phenomena for explanation. In a real sense we use the data as a ‘data model’ of the phenomena using the measuring process, as Suppes recognised:

*“It may be noted that in using an N-tuple as a realisation of the data ... we have taken yet another step of abstraction and simplification away from the bewilderingly complex complete experimental phenomena” [431] p.256*

The extended picture is shown in figure 5.

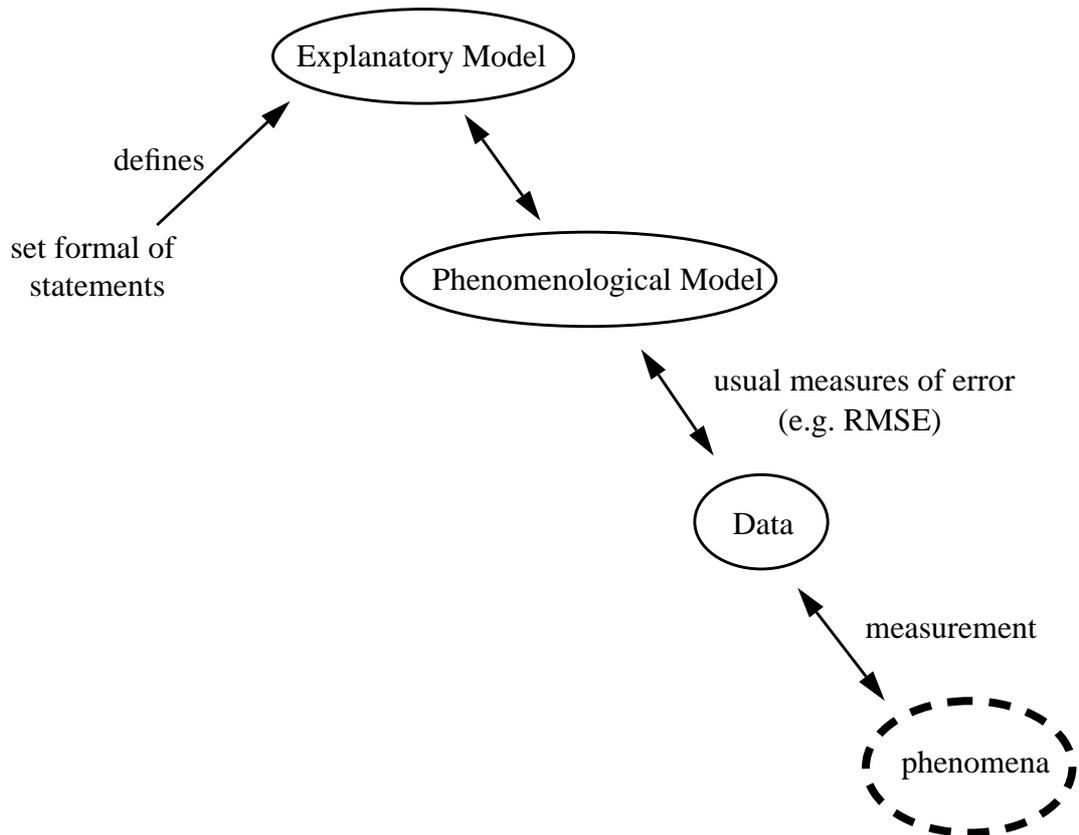
The reality of this distinction make it possible that observations may be theory-laden, since if our data was completely determined by the natural phenomena then the theory-ladenness would not effect our measurement.

Kuhn wrote:

*“Far more clearly that the immediate experience from which they in part derive, operations and measurements are paradigm-determined. Science does not deal in all possible laboratory manipulations. Instead, it selects those relevant to the juxtaposition of a paradigm with the immediate experience that that paradigm has partially determined.” [274]*

Cartwright emphasises rather the process of active adjustment that occurs during the preparation of data:

“...when we present a mode of a phenomenon, we prepare a description of the phenomenon in just the right way to make a law apply to it.” [84] p.157



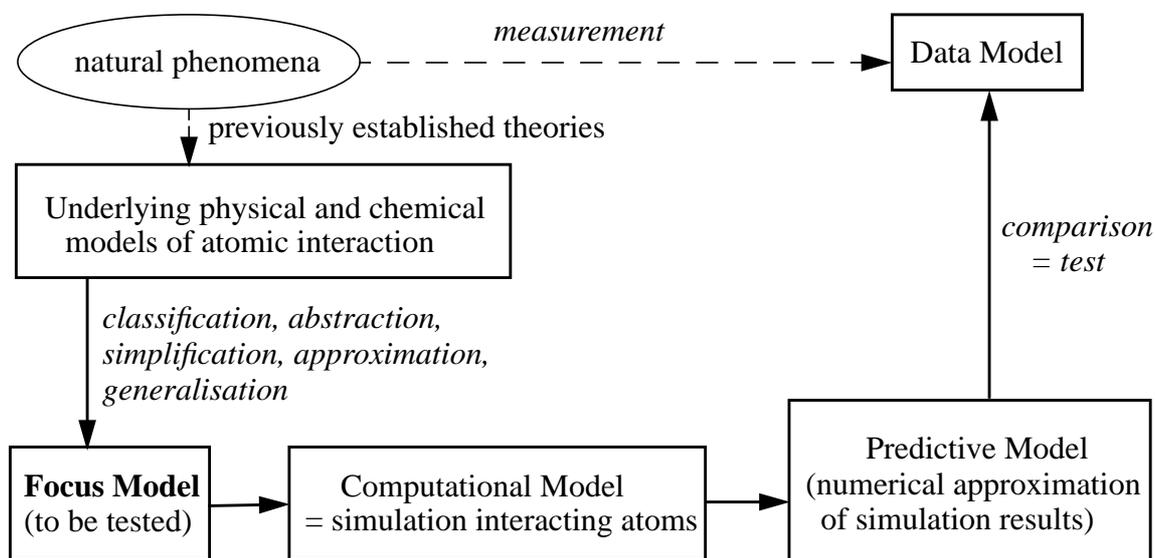
**Figure 5.** The picture with the data model added

I have built up a picture where a whole series of models come between the phenomena under study and the theory that is supposed to cover them. The relationships between these models are not always clear, especially that between the explanatory and phenomenological models. Cartwright argues that a ‘covering-law’ account is insufficient, as it frequently does not seem to be the case that one can derive aspects of the phenomena by gradually adding auxiliary hypotheses and facts as steps down this ladder [84]. Thus one comes to a picture of models as objects that somehow *mediate* between the phenomena and the theory - models as mediators [329].

For example, in chemistry the gap between theory and experimental phenomena is being increasingly bridged via the extensive use of computer-based simulations of many

interacting atoms [201]. A model of chemical interaction or structure is posited using approximation, simplifications, abstractions, etc. of the underlying (and previously well established) physical and chemical models of pairwise atomic interaction. This model is then encapsulated in a computational model involving many such interacting atoms. The simulations are run to explore the envelope of possible outcomes which are then approximated by mathematical functions. Thus the simulation substitutes for the intractable symbolic inference that otherwise would be necessary. Finally this post-hoc predictive model is compared against the data (or data model) derived from a measurement process applied to the chemical phenomena under study. This is illustrated in figure 6 (which is an adaption of Figure 17 on page 1021 of [201]). I quote

*“... the core of any model is the approximations, assumptions and simplifications that are used in its formulation. Only an understanding of the basics of a particular model may lead to sensible application or improvement of its performance. ... Due to the complexity of the systems of chemical interest, theoretical methods only became of practical interest with the advent and development of computers.” [201] p.1021*



**Figure 6.** The models used in chemical simulation (adapted from [201])

Giere [176] goes further than a mere hierarchy of models, and also posits the existence of ‘horizontal’, ‘vertical’ and local ‘radial’ links between models. In this picture a theory consists of a whole structured cluster of objects: laws, models and analogies. This

is surely the way forward: some of the confusion involving models in the past does seem to be due to the fact that many of the entities discussed as if they were a unitary structure are actually complicated composites with a series of different levels and processes.

This was foreshadowed by Suppes when he said in 1960:

*“... a whole hierarchy of models stands between the model of the basic theory and the complete experimental evidence. Moreover for each level of the hierarchy there is a theory in its own right. Theories at one level is given empirical meaning by making formal connections with theory at a lower level.” [431] p.260.*

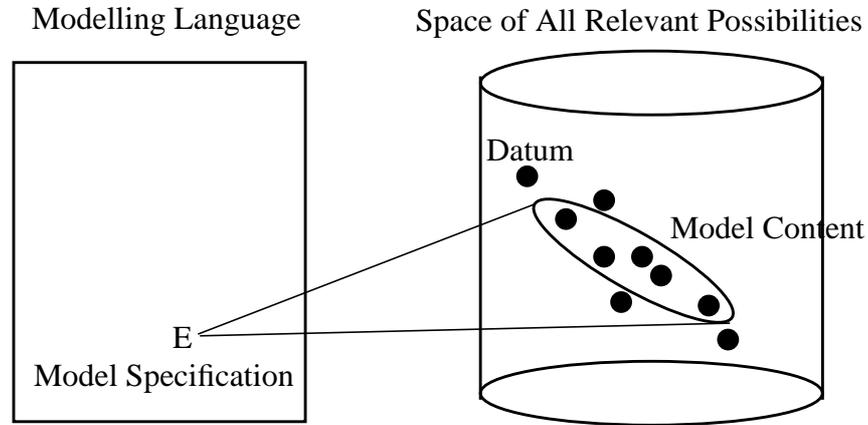
### **2.3 Parts of the Modelling Apparatus**

Given the above variety of definitions and deployment of models, I will attempt to ensure the clarity of terminology throughout this thesis by presenting a model of the modelling apparatus and then labelling the various parts. This will combine some of the best aspects of the philosophical analysis outlined above with a clear and unambiguous method of referring to the relevant parts and processes.

This picture of modelling broadly follows the model-theoretic approach, particularly as described by DaCosta and French in [129]. At its core is a state-space (following the approach of Beth and Weyl), which can be thought of as a systematic enumeration of all the possible data combinations that could result from the phenomena under study. So for the gas laws one might have a three dimensional space defined by coordinates for pressure, volume and temperature.

Different aspects of the modelling situation are all mapped into this one space. Thus the model description defines a subspace of these possible states, and it is this subspace which represents the model semantics. The data will be a series of points in the space and (in control situations there may also be a subspace representing the goals). The whole space can be circumscribed by relevant prior knowledge of the relevant possibilities (in the gas law case described above we know that we do not have to include temperatures below absolute zero). I will call the combination of model specification, model content and the relation between the two, the ‘model-apparatus’. This is illustrated in figure 7 (the situation is shown in a simple way for presentational reasons, in more realistic cases the

model content might not be continuous, connected, have a ‘hard’ edge (or membership function) or even be dense.



**Figure 7.** Model specification and semantics

The ‘modelling language’ is the set of possible specifications along with some relations between the specifications (equality, implication, etc.) and the way that these specifications are mapped onto the space of possible states. Following the model-theoretic approach the model content is the set of states which *satisfy* the specification. This modelling language could be a formal language, a subset of natural language, or even a real valued vector space. A good modelling language will not only be expressive enough to clearly specify the required possibility spaces, but also it will have explicitly defined relations that systematically reflect the corresponding relations between the possibility spaces. So, for example, the set-theoretic union of the possibility spaces might correspond to a disjunction in the modelling space. It is the fact that a good and expressive modelling language can accurately *reflect* relationships between the content of models that allows there to be syntactic as well as semantic accounts of modelling, but it is the non-syntactic approximation of models by their content (as described in [372]) which suggests that an account which includes the semantic side might be the more general.

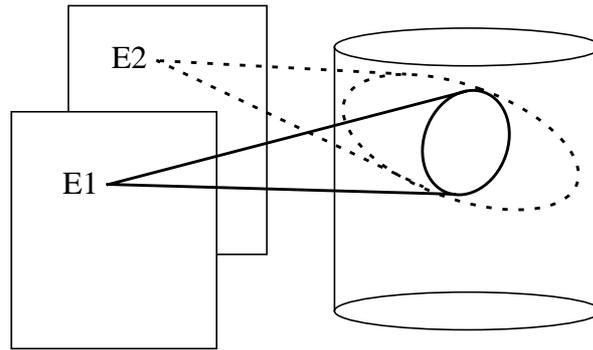
The set of data (which may not be points but small volumes or even fuzzy areas) form what I will call the ‘data model’. This data model is itself a specification of the phenomena in ways directly analogous to the relation between model specification and the possibility space. In fact measure theory is the formalisation of such a relation, where the ‘modelling language’ is a language of numbers, and the content is the phenomena being

measured, such that certain relations between the phenomena are homomorphically reflected in a relation on the set of numbers (e.g. concatenation and order).

The key idea (loosely following the approaches of DaCosta and French [129] and Redhead [372]) is that models are related and compared in the state space rather than via the model specification. A simple example of such a comparison is a measure of the error (for example the Root Mean Squared Error), which formalises the distance between the data model and the model content. In general a number of ways of judging the similarity and ‘distance’ between models is used. So, for example a phenomenological model content may *approximate* to a subspace of the theoretical laws’ content space, which may or may not be obvious from the specification of the phenomenological model and theoretical law. In other words it may not correspond to a consequence relation on the model specifications.

This picture suggests that there are at least three basic ways of relating model-structures: *firstly*, where one model content is contained in another (which I will call ‘subsumption’); *secondly*, where one model content is approximated by the model content of another (which I will call ‘approximation’); and *thirdly*, where the specification of one model is *described* by the specification of a second (which I will call ‘chaining’).

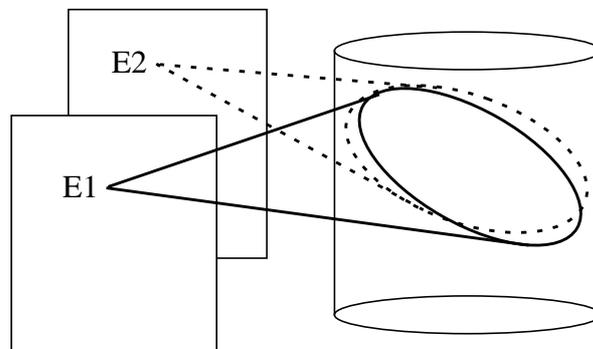
The subsumption of models, may be achieved by merely fixing some of the parameters of a more general model or law so that the restricted model only encapsulates some of the possibilities. For example the content of a general model of a damped and driven pendulum includes that of a free and frictionless pendulum. In general, the two model descriptions may be in different modelling languages, but mappable into the same state space. If the model specifications can be translated into the same modelling language and the inference relation on this language is complete (in the sense that every logical consequence can be proved using the inference relation), then this subsumption may be demonstrable as an implication in the modelling language. In general, however, this need not be the case. Relating models using subsumption is illustrated in figure 8.



**Figure 8.** Relating model-structures via the ‘subsumption’ of model contents

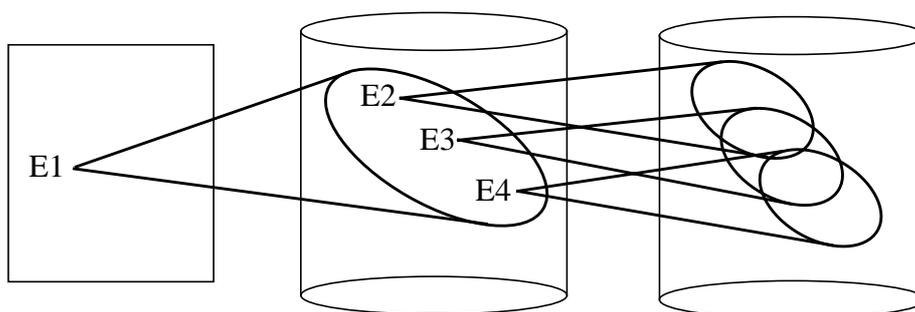
The approximation of model contents can be achieved in a variety of ways. One may attempt it by careful manipulation of the model specification, although one has to be careful about this since a small change in the specification can cause a large change in the model content. Alternatively numerical simulation of the results of one model can be used as data to fit the second model. An example of this can be seen in the process of chemical simulation mentioned above, where the results of the simulation are fitted to a numeric function – in this case it is the role of the computer simulation to enable the approximate inference of a model where the analytics are too onerous. Relating models by the approximation of their content is illustrated in figure 9. This is comparable to Redhead’s figure on page 151 of [372] in terms of relating model content. Such approximation of model content is implicit in the following quote from Cartwright:

*“...we lay out a model and within the model we ‘derive’ various laws which match more or less well with bits of phenomenal behaviour.” [84] p.161*



**Figure 9.** Relating model-structures via approximation of model contents

The relation of models by chaining is where the specification of one model-structure is used as the content for another. A example of this is how the data model specification (typically a list of N-tuples) is approximated by the content of a phenomenological model. This thesis is going to present a characterisation where model specifications are modelled by numbers, so that these act as a useful guide as to their complexity – which means that ‘complexity’ would be a (numeric) model of models of data models of phenomena! Relating models by chaining is illustrated in figure 10.



**Figure 10.** Relating model-structures via ‘chaining’

In the set-ups described above (section 2.2 on page 28), one arrow could actually represent a combination of several of these methods of model relation. The example from chemical simulation illustrates the use of all three of these types of model relation, as well as a combination of them: the data model and the predictive model are related by chaining; the predictive model is related to the computational model by approximation; the computational model is related to the focus model by subsumption (if it has been programmed accurately) and the focus model is related to the underlying theory via a combination of subsumption and approximation.

Note that I am not making a strong epistemic or realist claim here about what models and theories *are*, but merely using a simple model-theoretic structure as a building-block to model some of our constructs in an useful way. This is sufficient for my purpose here – I will leave the question of whether this reflects the actual use of models by scientists to others.

Different properties are usefully related to either the syntactic or semantic sides of the modelling apparatus. Thus the rate of error is a property of the model semantics whilst formal inference is more usually *attributed* to the model specification (even if, as a

consequence relation, it can be *defined* using the content subspaces). The falsifiability of a model is more usefully associated with the model semantics: to take an extreme example, if we look at the null model (‘anything may happen’), this would correspond to the whole possibility space – in such a case there is no possible datum which would disconfirm the model. The falsifiability of a model (or *specificity*) may be associated with a ‘volume’ measure on the space *not* in the model content (for more on this and some of the consequences of this picture of modelling see [333]). These properties are summed up in table 1.

model syntax	model semantics
context-independent specification, formal inference, (complexity!)	error w.r.t. data, approximation, inclusion, specificity

**Table 1:** Some properties ascribed to the model syntax and semantics

My thesis will be that ‘complexity’ is a property that is more usefully attributed to the syntactic side of the modelling apparatus, because of the existence of simplification and the importance of the modelling language to practical judgements of complexity.

## 2.4 Models in Machine Learning

Another field where models have been actively considered in their own right is that of machine learning and artificial intelligence. Here a model can be any internal construct of an (artificial) agent that helps it decide what action to take, though usually the term also has predictive and representational connotations. These models will often be induced from the data, so there may not be an encompassing theory (other than what is implicit in its own structure or representational system). Thus the term has far more of an ad-hoc and pragmatic flavour here.

Thus in contrast to the top-down elaboration of the philosophical account of modelling described above, is the bottom-up process that frequently occurs in machine learning. Here the question is not ‘Why isn’t the general formal theory adequate on its own?’ but ‘Why isn’t the simple data model enough?’. A number of answers can be given:

- We know from experience that data sequences are rarely repeated exactly;

- We also know that, frequently, factors irrelevant to our purposes will have affected our data (e.g. ‘noise’);
- The data model does not allow us to predict in circumstances we have not come across before;
- We sometimes have prior knowledge that allows us to make a good guess at a more complete set of possibilities without trying them all (e.g. interpolation in continuous processes);
- The data model is not a compact form of representation;
- It is difficult to ascribe meaning to a set of raw data;
- Direct formal manipulation of the data is difficult<sup>13</sup>.

All these lead in the same direction, namely towards some encapsulation as a specification in a more abstract (and maybe more general) model description via its (possibly approximate) encoding in a modelling language. The idea is to find a model description in a modelling language whose content closely approximates the data using what prior knowledge one has of the source of the data. I will claim that the *difficulty* in finding such a model specification is a useful characterisation of its complexity.

Given the nature of data gained by measurement from natural phenomena and the nature of descriptions in formal modelling languages, it would be very surprising if a compact or convenient description were found which exactly specified the data model – but, more fundamentally, it would not be very helpful in terms of generalising to areas of the possibility space were no data had been collected. Thus it is essential that the modelling language (and search process) is such that it does *not* result in a specification that directly describes the data. Almost always a model specification is settled on which either covers (i.e. includes) all the data points<sup>14</sup> or else approximates them to within an acceptable degree of error.

This is particularly sharply illustrated in the ‘over-fitting’ of neural networks, that is when the output of the network mimics the training data too closely and as a result (it is sometimes claimed) it does not perform well on new data from the same source. The

---

13. Though this particular difficulty has been significantly lessened with the advent of cheap computational power.

14. For example, in econometrics, a random distribution term is built into the model, so that the model will cover the data.

response is often to alter the structure of the network so that the space of functions that the network can produce is restricted, so the network is forced to abstract from the training data [401]. Of course, it is critical to the above strategy that the restriction utilises some prior and relevant domain knowledge concerning the nature of the data source, so that the network is restricted in an advantageous way, but this is often only specified implicitly (e.g. in continuity assumptions).

The last example highlights the importance of the modelling language to the modelling process. This is sometimes called the ‘bias’ in the machine language literature. Although the search for appropriate models is usually characterised as a process acting within a fixed modelling language, a more general picture is when both the modelling language and the choice of models are variable. Usually a change in the underlying modelling language is a more radical step, as such it has been associated with a change in paradigm as portrayed by Kuhn [274]. Toulmin associates it with the major advances in science:

*“The heart of all major discoveries in the physical sciences is the discovery of novel methods of representation, and so of fresh techniques by which inferences can be drawn – and draw in ways which fit the phenomena under investigation. The models we use in physical theories, ... are of value to physicists primarily as ways of interpreting these inferring techniques, and so of putting flesh on the mathematical skeleton.” [438] p.34*

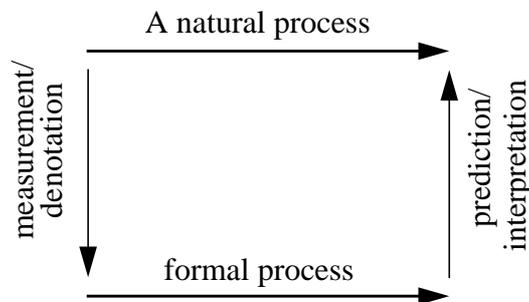
Later I will argue that a change in the modelling language is one of the most powerful ways in which simplification is achieved (section 5.7.4 on page 123). In machine learning the importance of the choice of bias and the possibility of dynamically altering it to improve the efficiency of the search process has been explicitly discussed (e.g. Russel and Subramanian in [393]).

Frequently a model is used in the active sense<sup>15</sup> that is it is used to predict some consequences given some initial conditions. So the model specification is converted into a process (this is clearest in the case of a computational model, when the model is encoded as a program and then run). This is particularly obvious in model-based control mechanisms where the error in the prediction of the model is used to adapt the model.

---

15. But not always, for instance a fashion icon may be taken as a model of fashion in a whole subsection of society.

To reflect the fact that, in use, models are dynamic objects, some authors prefer to talk about the ‘modelling relation’ rather than models (notably Rosen [385]). Here a formal inferential process is related to a natural process by encoding and decoding relations. The action of a natural process is modelled by encoding the initial conditions into the formal model, from which an answer is inferred by some symbolic process, which is finally interpreted back as a prediction about the result of the process. This can be seen as a dynamic version of Hughes’s DDI (Denotation, Demonstration, Interpretation) account of a model [236]. The modelling relation is illustrated in figure 11. In terms of category theory the formal process is said to model the natural process if, given the encoding and decoding mappings, the diagram commutes.



**Figure 11.** The ‘modelling relation’ from [385]

The connection between the former, static account of modelling and this dynamic account is the animation of declarative knowledge via the use of an inferential mechanism. Thus an equation is used to predict outcomes, by the substitution of variables and its solution, a program is animated by the program interpreter or processor. There are essentially process models without an explicit declarative counter-part (like a trained neural-network), but declarative model specifications tend to be more common. The prevalence of the declarative representation of knowledge is due to its flexibility: for example the same ideal gas equation can be used to either predict the pressure from the temperature and volume or the volume from the temperature and pressure.

## 2.5 The Philosophical Background to the Rest of this Thesis

My overall stance, from which I approach this thesis, is close to that of Giere<sup>16</sup> [176], except that I am not aiming to model science. My stance might be called a

pragmatic modelling stance in that what I am aiming for is a *useful model* of complexity. So, for example, when I am arguing what the property of complexity can be usefully attributed to (in section 3.3 on page 47), what is most important to me is to identify the most useful object for that attribution. I will argue that it is most useful to attach complexity primarily to the model specification and then allow for its post-hoc projection onto the model content and the target phenomena when the mappings to these are sufficiently constrained. These constraints could come from the nature of phenomena but could equally come from elsewhere, for example in the established methodology of a field. As Cartwright puts it:

*“It is precisely the existence of relatively few bridge principles that makes possible the construction, evaluation and elimination of models ... it strongly increases the likelihood that there will be literally incompatible models that all fit the facts so far as the bridge principles can discriminate.” [84] p.144*

My view on the constructivist-realist divide is that our models are at least somewhat constrained by the natural world, but not completely so. In some domains we may be more constrained by the world than in others: our model of a salt crystal as a cube may well reflect the world; our linguistic models of natural language may inevitably be socially constructed to some degree. As Giere puts it:

*“The real issue, as I see it, is the extent to which, and by what means, nature constrains scientific theorizing.” [176] p. 55-56*

Although the arguments herein follow more easily from a constructivist standpoint, they apply also to modelling as envisioned from a realist perspective.

To sum up: the aim of this thesis is to produce a general and useful *model* of the notion of complexity. The various existing formulations of complexity along with the other examples that I introduce play the part of the data. The main work is thus to find the model that matches these as far as possible but is also as ‘useful’ as possible. This will involve the formulation of the best ‘modelling language’ for its expression (i.e. one that matches prior reasoning about the domain’s nature), so that when we have finished our search process we end up with a specification of the concept that is not too complex but

---

16. Giere calls his position “naturalistic realism”.

adequately approximates our 'data'. In this regard, merely saying that every situation has its own complexity would be like sticking to the data model in the above, and this would be an inadequate account if we are ever to apply our account to new situations. As Einstein is supposed to have said when asked how complex a theory should be:

*“Everything should be made as simple as possible, but not simpler.”*

### 3 Problems and Properties

#### 3.1 Examples of Common Usage

I will introduce this section with some concrete examples, and recount some common sense perspectives on them. These will be referred to later and used as a motivation for the forthcoming discussion.

##### 3.1.1 A case of nails

It would be very unusual for someone, on opening up a large chest of nails, to exclaim “Oh, how complex!”. This indicates that the mere fact that there are a great many nails is, at least in some circumstances, insufficient for the assembly be considered complex (this would not prevent someone modelling the forces inside the pile<sup>17</sup> considering it complex).

##### 3.1.2 Writing a thesis

A thesis is easier to write with a complex tool (a word processing package with advanced features), than with a simple one (a ballpoint pen). This illustrates that the complexity and difficulty of a process can not be naively associated. Here it is the complexity of the *problem* that is associated with the level of difficulty, and the tool chosen forms part of the framework for tackling this.

##### 3.1.3 Mathematics

If anything is intuitively complex then abstract mathematics is. This complexity seems inherent in the subject matter despite the fact that questions of mathematics can be purely formal. Even when the mathematician has complete information about the rules and components of a problem, producing a solution or proof can still be very difficult.

##### 3.1.4 A gas

The macroscopic physical properties of a gas are fairly simple. Even though we know this is the result of a multitude of interactions between its component parts. If we had to explain these properties via an explicit and deterministic model (i.e. this particle collided with this one which...), this would be a formidable task. If we take as our base a

---

17. That this is a difficult modelling problem see the chapter in [91] on sandpiles

level of description that ignores a lot of the detail and ascribes this to an unanalysed randomness, then the task is considerably easier.

### **3.1.5 An ant hill**

In this example the interactions between the parts (i.e. the ants) are non-trivial, so an adequate model would probably have to include details on this. Now the task of explaining the macroscopic behaviour, given a model of the interacting parts, is challenging irrespective of whether the macroscopic behaviour is simple and predictable or not. This is the case, even if the ant colony looks very simple to someone else who views it as a rather stupid and reactive single animal that just happens to consist of physically separate parts.

### **3.1.6 A car engine**

Consider three views of a car engine<sup>18</sup>.

The first view is of the engine as a single inexplicable unit, a sort of glorified random number generator - it either works or it doesn't. No explanation is required or deemed relevant, its running is a matter of irreducible luck. If it does not start you ring a mechanic who magically starts it for you. This is the engine as a simple, if malevolent, entity.

The second is a view involving a partial knowledge of the engine. The parts are roughly identified as well as some of the interactions. However these interactions, taken together, are far too complex to understand. If something goes wrong, you can look inside the bonnet and try to identify the cause. Simple experiments in terms of fixing it are possible. Sometimes, with luck, this seems to fix it. Unfortunately, this action often has unforeseen consequences and causes greater long-term damage. When fixing it is beyond this level of understanding, the mechanic is called, who must be (from this viewpoint) a craftsman of deep skill and have a sophisticated understanding of the machine. This is an engine at its most complex.

The third view is (hopefully) that of mechanics. They have an understanding of the decomposition of the engine into functionally near-independent parts. They can use this model to systematically analyse the problem by eliminating possible causes, until the search narrows down to the actual cause. They can then fix this cause, having a good idea

---

18.This subject was suggested by my supervisor as a pertinent example, John Chidgey.

of the possible side-effects of this action. This is the engine as manageable complex, due to the appropriateness and utility of the mechanic's model of it.

### **3.1.7 A cell as part of an organism**

When considering a cell and the organism of which it is part, from the same frame of reference, it seems obvious that the cell is simpler than the organism. This is true irrespective of the complexity of the cell, for other cells can not take away from the complexity already there in the cell.

Such a comparison is not so obvious if both are compared from within different frameworks. A cell under a microscope might seem much more complex than the potato it was taken from, viewed with normal vision.

### **3.1.8 Computer programming**

As with the *writing a thesis* example in section 3.1.2 on page 44, the choice of programming language can greatly effect the complexity of a programming task. A language like BASIC may be relatively easy to learn, but difficult to use for a large programming task. Compare this to a language like Smalltalk which has a sophisticated object-orientated organisation allowing a high degree of modularisation and flexibility and a large set of pre-programmed classes that you can adapt and re-use. This takes considerably more time to learn, but can then make the task of programming much easier.

## **3.2 Complexity as a Comparison**

In the common sense world complexity is not so much measured as compared. “a computer is more complex than a calculator”, or “although the rules governing unemployment benefit are more complex than those concerned with income support, those covering disability benefit are the worst” are two examples of this.

Such comparisons are easiest to see when you are comparing a sub-system with the whole, from within the *same* frame of reference, as in the cell example above (section 3.1.7 on page 46). In other cases it is not at all clear, as when comparing the complexity of a computer and a poem. Here we have no natural common basis from which to make a comparison. We could artificially construct such a basis but there would be little likelihood that this would agreed upon by others. Without an appropriate framework

within which to represent both, any such judgement of relative complexity would be arbitrary.

### **3.2.1 The emergence of life**

One paradigm of emergent complexity is the appearance of life. Most people would say that the complexity of the biosphere has increased with the emergence and development of life. That is, if one compares the solar system as it is now and how we think it was 5 billion years ago, then the obvious conclusion is that it has become *more* complex in important respects. The fact that the subsystems which exhibit the change are not great in terms of mass or volume does not significantly alter this judgement.

## **3.3 What the Property of Complexity Could Usefully Refer to**

In this section I will argue that regardless of whether our models reflect reality in other respects, complexity is most usefully attributed to the *descriptions* of our models and is only projectable back onto natural phenomena when constraints on our choice of models make this coherent.

One could attempt to distinguish what complexity was an *intrinsic* property of, and thus argue that complexity was an extrinsic property of natural systems but an intrinsic property of model descriptions. Such a route is fraught with difficulties and would not further the purpose of this thesis, which is pragmatic. I am concerned here with developing a characterisation of complexity that *usefully* captures our intuitions of the concept.

### **3.3.1 Natural systems**

I will argue that complexity is not a property *usefully* attributed to natural systems. I list the arguments below.

1. *Estimates for the lower bounds of the complexity of natural systems can always be increased by the inclusion of more aspects of the system*

By increasing the level of detail considered, a lower bound for the complexity of almost any natural system can be arbitrarily increased until it is well beyond our means of representation, understanding and modelling ability. There is no apparent upper bound to how complex things can appear. It seems that most things have the potential to be

arbitrarily complex, just dependent on the number of aspects and the level of detail one is willing (or able) to consider.

This can be seen as a consequence of our intuition that a sub-system is less complex than the whole. By expanding the context of consideration (or equivalently using a more general framework) one includes more complex sub-systems. This forces us to conclude that the complexity of the whole system is greater than these sub-systems (unless one is allowed to ignore the effects of these sub-systems using appropriate modelling assumptions).

For example, a brick is a fairly simple object if you consider only its macroscopic properties but is much more complex at the level of its component particles. This particulate complexity is averaged out at the macroscopic level at which we usually relate to them, so we can usually abstract from the details, but if you are insisting on an entirely objective basis (what ever that would mean) then you have to allow for the inclusion of this particulate complexity. The complexity further increases beyond our comprehension when we consider such a macroscopic object, such as a brick, at the sub-atomic level of detail.

2. *Estimations of the practical complexity of natural systems can change critically when the framework in which they are considered is varied*

Changes of goal, language of representation, aspect, and scale can all greatly effect the practical complexity of natural systems. The point above is a result of the potential to include more detail by changes of scale and generalising so as to include more aspects of the system under study.

In the ant colony example (section 3.1.5 on page 45) whether your goal was merely predictive of behaviour at a purely macroscopic level or was seeking to explain this macroscopic behaviour explicitly in terms of the behaviour of individual ants, affects the practical complexity of the modelling task.

One example of how the language of representation can critically affect the complexity is in the representation (or exclusion) of elements as noise. Often ascribing some parts of some natural phenomena to noise can allow the drastic simplification of the representation of a natural system. For example in the gas example above (section 3.1.5 on page 45) being able to assume that the detailed movements of the particles in a gas are

random allows a model of the macroscopic properties to be related in a simple way to the microscopic ones.

Thus intuitive assessments of the complexity of systems, often differ far more with changes across frameworks than across changes of subject matter.

3. *Attributing complexity to natural systems does not help explain the existence or process of simplification*

The ascription of complexity directly to natural systems also makes an account of simplification difficult. One would be forced to judge all equivalent models of a natural system as equally complex. Thus an account of planetary orbits using an infinite series of epi-cycles would be as simple as one using ellipses. Similarly there would be no simplification of mathematical systems if complexity was a property of the *content* of these systems, since all we would be doing is changing their description.

One method of simplification (discussed in section 5.7.5 on page 124) is to trade-off complexity for the specificity or accuracy of the model. A less specific model is one which has either narrower conditions of application or else its predictions are over a broader range – for example, the use of fuzzy-logic and fuzzy-set theory has been suggested as a means for dealing with complexity in some situations by Zadeh [488]. A less accurate model is one with a greater level of error with respect to the data model – for example, one might decide that some variation in the data could be attributed to noise so that accepting a greater level of error might result in a much simpler model. The ‘stochastic complexity’ of Rissanen [378] is an attempt to find a principled trade-off between error and model complexity.

In fact simplification and elaboration (for the want of a better antonym) are frequently what we are concerned with when we talk about complexity – complexity has somehow arisen and we need to deal with it. Some strategies for simplification are discussed in section 5.7 on page 120.

4. *Our intuitions about the complexity of natural systems can be nicely accounted for by associating them with the complexity of the systems ‘best’ model*

If we have a natural system which is producing what seems to be random data as its output (where we know that this is not attributable to a separate and discountable source of noise) it can still be sensible to say that this data is simple on the grounds that it is random.

Here the complexity of the system has been taken from its most appropriate model, which in this case is *not* the most descriptively accurate but is a less specific model. A probabilistic model is more appropriate in circumstances where we know the detail of the individual sequence is not relevant. We lose the possibility of completely accurate predictions for a considerably simpler model.

In cases where our models of these systems are considerably constrained (by nature or by practice) we are sometimes in the fortunate position of only having one candidate model in which case it is safe (in that context) to project this model's complexity upon the natural system. As Cartwright puts it:

“It is precisely the existence of relatively few bridge principles that makes possible the construction, evaluation and elimination of models ... it strongly increases the likelihood that there will be literally incompatible models that all fit the facts so far as the bridge principles can discriminate.” [84] p.144

One of these principles is surely that we exclude models that are prohibitively complex. Reinberger recently said:

“Reduction of complexity is a prerequisite for experimental research.” [374]

All of the above difficulties come down to the same nub: if natural systems do have complexities, then they are unmanageably large. Thus at the moment, and quite possibly absolutely, it is not useful to try to do this. If we choose only one aspect and one scale, we are no longer dealing with the complete object, but an abstraction of it.

This can be traced to why we *ever* consider properties to be of things rather than our models of them in the first place – because they can be said to have the “same” properties independent of the observer or the models<sup>19</sup> (e.g. mercury or thermocouple models of temperature) - there is no evidence that this is true of complexity judgements.

There are several possible arguments against this. I consider these below.

1. *Some complexity comparisons concerning natural systems are objective.*

An example is “An amoeba is objectively simpler than a human”. There are two ways of interpreting this argument, firstly as a variant of the cell example (section 3.1.7 on page 46), i.e. that an amoeba is simpler than a human *however you look at it*, and secondly

---

19. This is a simplistic account – accounting for why we call a property the same from different frameworks is not straightforward.

as an implicit call on upon a privileged (and hence common) framework, i.e. there is *one* sensible framework to use and within this the amoeba is simpler.

The first argument assumes that the comparison is valid, *regardless* of the framework. This is an immensely strong assumption, one that seems to draw its strength from an identification of the amoeba with a human cell<sup>20</sup> and invoking the sub-system property that I noticed in the cell example. Otherwise it would seem possible to choose a framework where the amoeba differed substantially from the cell (its method of encapsulation and subsequent digestion of food?), where the judgement of relative complexity was not so clear. If one then argues that this is an inferior or more specific judgemental framework, this would bring us to the second interpretation.

The problem with arguing for a uniquely (or even relatively) privileged framework is that of its justification, given that it does not allocate impractically large complexities to almost everything (or that a framework revealing *more* complexity is not always better). Also that in practice there are always pragmatic choices of factors like scale, so that a privileged framework is no use for actual complexity judgements. Finally the very identity of many things seems inextricably linked to which aspects you are considering (e.g. ‘society’).

2. *Claiming that complexity is not a property of natural systems is just a category mistake.*

It could be argued that even if it is admitted that complexity comparisons can only be meaningfully via our models, the complexity refers to the natural systems themselves, and that it is only due to our limitations (in particular our understanding) that we see different complexities from different viewpoints.

This is a possible standpoint, but it is hard to see how this could be then applied in practice without *in effect* attaching the property of complexity to the models rather than the original systems, as otherwise in practice the complexity of such systems would change arbitrarily depending on the model chosen. Such a view would be comparable to that attributing primality to sets of real objects rather than numbers. One could characterise number by equivalence classes of things, i.e. five is the class of all possible

---

20.Or alternatively an identification of the functional organelles of an amoeba with the corresponding functional parts of a human, though this does not alter the argument.

sets of five objects. This still would mean attributing a property such as primality to something other than the things themselves. It is difficult to see this as anything other than a contrivance.

3. *Attributing complexity only to model descriptions would make objective complexity judgements impossible.*

If a framework is agreed upon then the complexity of something can be objectively determined by different observers with respect to this framework. So once this framework is established complexity judgements can be consistently made irrespective of who is doing it as long as they keep within the rules that the framework entails. This is not so different from many other ‘objective’ judgements and facts – such frequently rely upon such contextual bases (what Suppe calls the ‘disciplinary framework’ [428]).

For example in the physical and mathematical study of chaotic systems a framework which implicitly disregards some level of detail as noise is so familiar that it has become a background assumption (see the section on Noise on page 206). Researchers in this field seem to uniformly agree that a perfectly disordered system is simple (like the gas example, section 3.1.4 on page 44, where this uniform randomness makes it a candidate for a level of simplicity akin to a crystal<sup>21</sup>). Of course, it is advantageous to make these background assumptions explicit, so that if necessary they can be checked.

4. *There are real causes of complexity*

This may be true but not a compelling argument for the use in considering complexity as pertaining to natural systems. It derives from a confusion between accounts of how we are to characterise complexity and what can *cause* it. To take an analogy heat can be *caused* by a variety of forms of energy which *are* themselves not represented as heat.

In the end pragmatic considerations prevail; it is *useful* to attribute a property to a natural system when this is largely independent of our models of it but more useful to consider it a property of the models if it is not (e.g. beauty is partially attributed to the

---

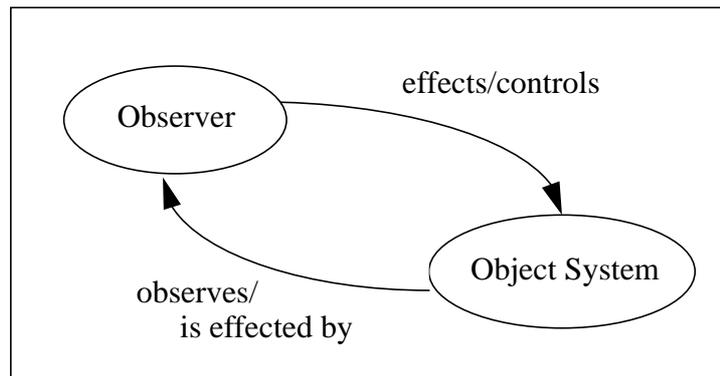
21. For a classic account of this see Grassberger [194].

beholder, because is known to be at least partially culturally dependent - 60's tower blocks - Miss World competition). As Suppes said:

*“We can only hope to have a concept of complexity for explicit models of the world and not for reality itself or even small parts of it.” [433].*

### 3.3.2 The interaction of an observer with a system

In response to these problems many authors (e.g. Casti [87]) have stressed that complexity only makes sense when considered as relative to a given observer<sup>22</sup>. Thus they put an observer into the picture which controls or otherwise affects the target system, and then observes (or is effected by) that system. This establishes a split between the “system complexity” and the “observer complexity” (see figure 12). The system complexity is the complexity of the system w.r.t. the observer and the observer complexity is the complexity of the observer w.r.t. the system.



**Figure 12.** Observer–system pair

You can look at this analysis in two ways: extrinsically and intrinsically. In the first, there is no essential difference between the two systems - from an external point of view one just sees two systems interacting. In the second, we are describing the situation from the observers point of view.

If you take the extrinsic interpretation, then there is nothing special about the observer system. The ‘loop’ in the diagram (figure 12) is misleading as there may be multiple parallel and re-entrant interactions between the two systems and between

---

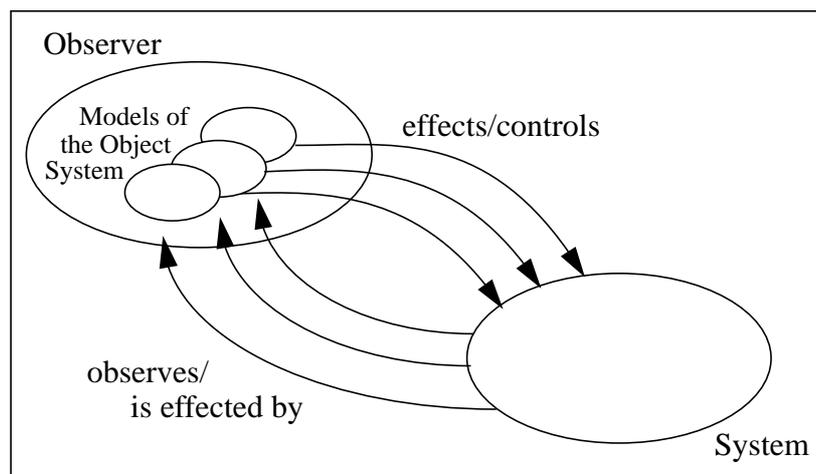
22.This distinction is basic to systems theory, going back to Ashby [23, 24].

different sub-systems of the observer and object system (as in figure 13 on page 54). In this case we are merely dealing with one system composed of the pair of object system and observer. The observer and object may or may not be easily separable or have effectively separate identities. All we are dealing with is a particular decomposition of a single system. So the observer and system complexity become merely the complexity of a sub-system with respect to the rest.

If you take the intrinsic view then there are still some problems with this approach, namely:

1. *It is still difficult to ascribe useful meaning to the complexity of the observer w.r.t. the system unless the system is an observer too, otherwise the observer is itself unobserved and so the complexity undefined in the same way as with natural systems (section 3.3.1 on page 47).*
2. *The complexity of the system w.r.t. the observer will still vary according to which aspect of the observed system is being considered by that observer.*

The observer could be taken as to refer to a particular identified individual. This individual could use several different internal representations to model the system and interact differently according to each at different times. If these act in some way such that they can be considered together as a composite model then you are back in the situation illustrated in figure 12 on page 53. Such a scenario is illustrated in figure 13.



**Figure 13.** An observer-system pair with multiple models

Either the separate models can be said to be interacting with the system (through the observer) or the internal models can be said to form some sort of a composite model. If, in the former case, the separate models would not qualify as separate observers, then problems of complexity being changeable according to the particular viewpoint chosen reappear. In other cases the situation can be further abstracted as to be between the model/representation and the system.

So which ever way you interpret the observer-object system analysis, you come back to the same problem of practical reference to an embedding framework drastically effecting the effective complexity of the observed model.

### **3.3.3 Patterns**

Next I will argue that complexity is not a useful property to ascribe to patterns<sup>23</sup>. In other words, to make a meaningful judgement as to the complexity of a pattern you need a syntax.

Consider two patterns generated by a random process; how can you judge them as differently complex? One may seem to be more meaningful and the other not but this is maybe just happenstance, it may be that they are both equally probable and generated by the same process. It is only through the interpretation of some process that they may be said to differ (for example by compression to a minimal length Turing machine that would output it, see section 8.2 on page 136). The trouble is that the same pattern can be decomposed or interpreted in many different ways. Each decomposition might give a different picture of its complexity. In Appendix 2 (section 9.1 on page 164) I show that given some reasonable assumptions that there is no non-trivial complexity measure on one-dimensional patterns. This counter-example relies on the possible multiplicity of decompositions - it contrasts with the demonstration of complexity measures upon a structured language given similar assumptions.

The usual manner for dealing with the complexity of patterns is not to compare the patterns themselves, but to compare the respective sets of rules for generating these patterns (even if these rules are merely guessed at). This explicitly provides them with a syntax, and hence they become more than a pattern.

---

23.By a pattern, I mean data that is ordered (typically in time or space) but is not restricted by a combinatorial syntax.

### 3.3.4 The modelling relation

Rosen [385], Casti [89] and others, have formalised the process of modelling in terms of a “modelling relation”. This was explained in section 2.4 on page 38. The intention is that a chain of causation in the natural system is modelled by first encoding the initial conditions into the formal system, then following the chain of formal causation and finally mapping this back into the natural system. This is illustrated in figure 11 on page 41.

If the modelling relation commutes the formal system is said to be a *model* of the natural system. Rosen then characterises “complexity” as an attribute of a natural system, if there are many such “inequivalent” models (see section 6.7 on page 131). Casti quantifies this as the number of inequivalent models (section 8.29 on page 152).

In order for there to be many such models, there must be many possible encodings of the natural system. In order to allow for the existence of several such encodings, the whole must lie within a larger framework. In order for “inequivalence” to be well defined, this larger framework must *itself* be sufficiently defined. Finally, this framework must be limited in scope, otherwise all natural systems would trivially have many models and thus be “complex” in this sense, in which case it would not be a useful property *of* such “natural systems”, as it would be coincident with the property of *being* a natural system. It is unclear in this case that *any* natural systems would be counted as simple by this criterion.

Thus, in order to be meaningful, this approach to attributing “complexity” to natural systems, must be defined relative to a larger framework. To the extent that this framework is well-defined, then this approach will be also. Some of the advantages and difficulties of this approach will be discussed in section section 4 on page 72.

### 3.3.5 A model with respect to a specified framework

In view of the above analyses I contend that complexity, if it is to be a useful attribution, needs to refer to a model relative to the modelling framework. That this is a possible or useful approach will be demonstrated in section 4 on page 72, where I will give a working definition of complexity and apply it to some examples.

### 3.4 Some Unsatisfactory Accounts of Complexity

Here I will briefly discuss some ideas that are frequently conflated with complexity, but which are, at best, very weak models of it. In each case conflating the concept with complexity will mean the loss of a useful analytic distinction. In doing so I hope to motivate some of the details of the next sections concerning the desirable properties of a complexity measure and its definition.

#### 3.4.1 Size

There is clearly a sense in which people use “complexity” to indicate the number of parts, but it seems rarely used *just* to indicate this, as was shown in the case of nails example (section 3.1.1 on page 44). Contrast this example with that of an intricate (mechanical) watch, where the appellation of complex might be more appropriate.

The difficulties raised by size are real, but can be weak when compared to other such difficulties; it is likely that simple problems such as size can be dealt with in simple (albeit possibly expensive) ways<sup>24</sup>.

Intuitively there are large but simple systems, such as a book of random numbers, or the list of facts behind ‘Trivial Pursuit’ questions (where there is no intended relevance between the questions other than the categories of sport, entertainment etc.). Doubling the size of either would not significantly increase their complexity.

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, it is very hard to imagine anything complex made up of only one part. This minimum size can be quite small, there is a Turing machine defined with only five states that comes to a halt after exactly 23,554,768 steps and the task of finding the maximum number of steps a seven-state Turing machine could definitely halt in has been described as “hopeless” by Machlin [303].

If a system is broadly symmetrical in terms of the relations between its parts (for example in a peer organised computer network), then size might be a good indicator of complexity, but otherwise the structure of the system might have a far more critical effect.

---

24. This is not to deny that if there a size limitation is a critical factor this may not qualitatively change the situation to a complex one (as Anderson [13] points out).

Broadly size seems to limit the potential for complexity, rather than determine it. See section 8.38 on page 157 for examples of size used as complexity.

### 3.4.2 Size of rules

A variation on a purely sized-based measure would be to base a measure on the size (in some sense) of the *description* of a system (in some language). This produces far more acceptable results, for example the book of random numbers would be judged in its complexity by the size of rules that generated it, rather than the size of the results.

However this still has many of the same problems as simple size measures have. One can imagine a case where a system was generated by hundreds of independent rules (e.g. 1001 unrelated tips about how to succeed). Thus the interaction between the rules must be still important as to the overall complexity. Of course, it is hard to imagine examples in natural language where there is absolutely no relevance relation between separate descriptions, due to the involved and intricate nature of language. This does not stop it being the case that the amount and structure of this interaction or relevance between the parts of the description will effect the complexity of the result.

One can imagine further elaborating this sort of approach by then taking the description of the description of the pattern and thus further improving this way of measuring the complexity of the original object. This improves the suitability of this sort of measure for complexity in that it encodes more about the structure inherent in the object, but in this case one is abstracting further and further from the basic model of size as a measure. The complexity of the original object is now far more encoded in the language of description than the eventual associated size. On the whole it seems that the more a judgement of complexity encodes *structure* rather than mere *size*, the more it fits our intuitive picture of complexity. This puts size measures in their place, they encode only the most basic property of structure - the number of parts the structure is to be built from. Thus if simple size could be seen as a measure of complexity it would be amongst the weakest possible of such measures. See also section 8.39 on page 157 on size of grammar for examples of this approach.

### 3.4.3 Minimal size

This brings us to a further elaboration of size as a complexity measure: that of the *minimal* descriptive size in some language or by some compression process (see also

section 8.24 on page 149). The most commonly used form of this is “Kolmogorov” or “Algorithmic Information” complexity (AIC) - the size of the smallest Turing machine that would generate the object pattern (see section 8.2 on page 136).

AIC represents the high end of minimal descriptive size measures - those using the most powerful compressive machinery (alternatively those using the most expressive language). The compressive machinery is so powerful that it gives a highly unpredictable measure. For example the second 1,000,000 digits of Pi are indistinguishable to most people from a list of 1,000,000 random numbers, but the first would have a small AIC and the second would (almost certainly) have a large AIC<sup>25</sup>. Here the relation of the size of the minimal program and the object are only marginally related - a fact that is starkly illustrated by the fact that the AIC size is uncomputable, in general, from the original object. I argue later (section 4.3.4 on page 84) that AIC is a more appropriate measure of information.

At the other end of the scale, if the compressive machinery is very weak (the associated language is inexpressive), then this is not much better than a measure based on simple size, since the size of the object and the size of the minimal description will be highly related.

Measures based on minimal descriptive size could be an acceptable complexity measure if the relevances between the parts of the original that the compressing machinery exploited (i.e. the redundancies) were the appropriate ones for the system and purpose in mind. In any case this is now primarily no longer an issue of size but structure.

### **3.4.4 Processing time**

It is hard to imagine a difficult task that can be done without some time spent on it, either in execution or preparation. Thus the complexity of a task can come to be associated with the amount of processing time it requires (see section 8.47 on page 162 for examples).

That this is a probable consequence of complexity and not a sufficient condition for it can be seen in the existence of intuitively simple but time-consuming tasks. Trying out all the possible colorations of a checker board of a certain size, is a simple task, but given  $m$  colours and a board of size  $n$  there are  $m^{n^2}$  possibilities, so trying out the colouration of

---

25.It is overwhelming likely that the AIC of a random string is not less than the length of that string [284]

a 20 by 20 board with 10 colours would be far above the maximum possible computation performable by the whole universe since the beginning of time [75].

Secondly, like size, if processing time is a measure of complexity (see section 4.3.3 on page 84) then it is a fairly weak measure, relatively unreflective of the structure of the task. Processing time may sometimes be an insuperable difficulty in practice but it is a simple difficulty, solvable by simple means (more time)<sup>26</sup>. Personal experience tends to suggest that the process of writing a program often presents a more fundamental level of difficulty than the time it takes to run.

Like size the application of processing time as a measure of complexity can be applied at a more abstract level, for example, to the time taken to write a program to preform a certain task. This brings this sort of measure closer to the intuitive meaning of complexity, but at the cost of the direct appropriateness of the “processing time” measure. In the above example, in order for the programming time to be well-defined, the programming environment and methods would need to be specified, then the complexity of the task is more encoded in that environment than in the programming time taken.

### 3.4.5 Ignorance

Complexity is a major cause of ignorance - if a problem is complex, we often do not know how to solve it. Ignorance is sometimes a cause of complexity - due to our ignorance we can choose an inappropriate framework for considering a problem and this makes solving it more complex. It is thus tempting to associate the two. I will argue that information (or the lack of it) and complexity are more usefully considered as different aspects of a problem.

Firstly, there are some tasks where we seem to know everything, but they are still complex. This depends upon your scope of “everything”. In many puzzles and games one has complete knowledge of the rules and situation, but finding the solution is still complex. If one broadens the scope to include knowledge of the solution (or optimal playing strategy), finding the solution may be no longer difficult, but implementing it may be. Some games and puzzles have no *easy* solution (e.g. chess).

Secondly, there are situations where increasing the knowledge about the solution does not make it simpler. Imagine a task of designing the best program to check the code

---

26.This does not prevent questions *about* processing time being complex.

of another program to see if it will halt, given a particular input. We know that there are no techniques that do this in general [443]. Knowing this does not make the task less complex!

So although ignorance and complexity can interact<sup>27</sup>, they should be considered as separate sources of difficulty - requiring different types of solution. Separating these two factors will also allow the study of their interaction - which could be very useful. See section 8.15 on page 144 for examples of this approach.

### 3.4.6 Variety

Simple systems themselves do not display great variety<sup>28</sup>. Thus complexity can be associated with variety. An animal with a greater variety of shapes of invertebra is said to be more complex (in this respect) than one with fewer (e.g. McShea [316]).

An immediate increase in variety, can however, be accompanied by a decrease in complexity. A list of prime numbers up to a certain size is not less complex than a list of all such natural numbers, despite the fact that there must be at least as much variety in the list of natural numbers by construction. It is implicit in this comparison that it is not the sequence itself that is really being considered, but the rules/source of it - thus the rules to generate prime numbers are more complex than those for a sequence of all numbers. So the application of variety shifts to these rules, but this does not completely avoid the problem, as a base of *all possible rules* might not produce very complex behaviour.

Another possibility is that you measure variety by minimal description length described above (section 3.4.1 on page 57), but then the arguments of section 3.4.3 on page 58 would apply. See section 8.48 on page 163 for examples of this approach.

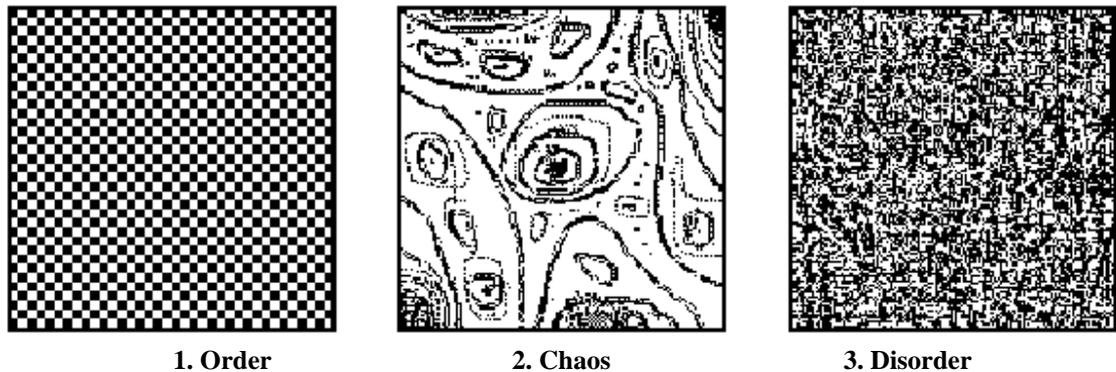
---

27. See, for example the “problem complexity” of Waxman in [463].

28. Although categories (or systems) of simple systems do.

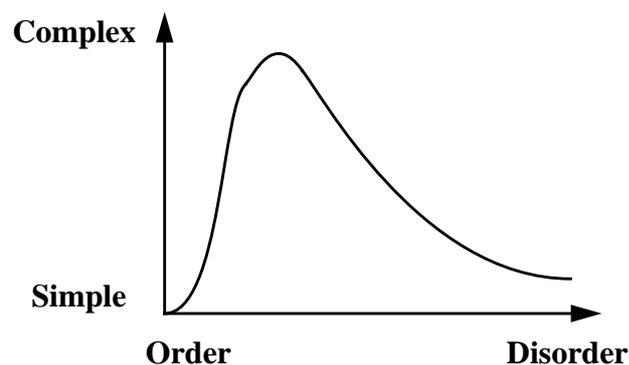
### 3.4.7 Midpoint between order and disorder

Complexity is sometimes posited as a mid-point between order and disorder. Grassberger [194] considered three patterns similar to those in figure 14 below.



**Figure 14.** Complete order, chaos and complete disorder

The immediate reaction is to judge that the first and last patterns are simple and the middle one relatively complex (leading to diagrams such as figure 15 on page 62, e.g. in [297]), but this is due to the facts that our perceptions ‘filter out’ the complexity of the right-hand pattern and that we interpret it as *representing* a situation with *no rules*<sup>29</sup> (i.e. random). Thus, we are not judging these uniformly; it may well be that the right hand pattern represents *such* a complex situation that we don’t recognise it. To illustrate this, consider the possibility that there may be a small version of the left-hand pattern included in the middle and a small version of the middle pattern included in the right-hand pattern (as in figure 16 on page 63).



**Figure 15.** Presumed graph of disorder against complexity

---

29.Or a single rule involving randomness. Whether this would be a complex rule would depend on whether you judge randomness as fundamentally complex or whether it is an allowed atomic rule form.

If this is the case, we are forced to judge the patterns in order of increasing complexity from left to right. We see the importance of the language of representation. If we were considering the complexity of some (assumed) rules to generate these patterns, then the original intuitions might be preserved. As noted in section 3.3.3 on page 55 the confusion comes because such patterns do not have an inherent language<sup>30</sup> - we have to impose one on them.



**Figure 16.** Possible diagrammatic inclusions

A footnote of Grassberger's is particularly revealing here. It says:

*“Some people hesitate between the middle and right panels when being asked to point out the most complex one. But once told that the right one is created by means of a (pseudo-) random number generator, the right panel is usually no longer considered as complex.” [194] (page 491).*

“Not being completely ordered” and “not being completely disordered” may be necessary conditions of complexity, but this gives us no way of comparing intermediate cases which are differently structured. The degree of variety and constraint involved in a system is only a rough measure of a systems structure, and thus is not an ideal measure of complexity.

### **3.4.8 Improbability**

In many physical systems (particularly systems in equilibrium), complex behaviour at a macroscopic scale is very unlikely; a uniformly disordered state at the microscopic level is normal.

---

30.Except maybe an assumed one, which may be different for different observers.

There is a marked contrast here between the different levels of description. A simple, uniform equilibrium state at the macroscopic level hides very complex and highly disordered detail at the microscopic level. Conversely, more complex behaviour at the macroscopic level is often based upon a more ordered state of affairs at the microscopic level because a disordered state (high entropy) tends to be very stable.

In terms of information the most efficient coding that can be devised will use less information to specify or record probable events than unlikely events. In this sense a probable event gives you less information because you expected it anyway [408].

Thus a high probability macroscopic state coupled with a highly disordered (high information) microscopic state is associated with low complexity and a low probability macroscopic state with an ordered (low information) microscopic state with high complexity. Thus entropy and other probability based measures are linked with complexity.

There are also low entropy states associated with low complexity, like cold perfect crystals, so the connection between entropy (of some of the other probability/information based measures) and complexity is not straightforward, as shown by Li [287]. In some highly dissipative systems with noise, however, complex behaviour is almost certain and a lack of it would be very surprising. An example is convection in a fluid between two parallel plates with a sufficiently great temperature gradient between them [361]. Thus the association of complexity with improbability only holds for a restricted set of equilibrial systems.

Part of the problem here is in the implicit determination of the appropriate scale for the phenomena to be viewed at, i.e. what level of detail is considered meaningless “noise” (see the section on Noise on page 206). The improbability of a complex state is entirely dependent on this scale, for without such an approximation every state is equally probable and meaningful.

Finally, complexity measures based on probability do not directly apply to purely deterministic systems without some form of course graining applied to them as above.

See section 8.22 on page 148 for examples.

### 3.4.9 Expressivity

If a type of statement can express a lot of things, i.e. it has expressive power, then it is likely to contain some more complex statements than a type which has a more restricted range. It is certain to include more such complex statements than a sub type could. You can not describe some complex phenomena without sufficient expressive power, and the theory of more expressive types is, in general<sup>31</sup>, more difficult than the theory of less expressive types. So an indication of the complexity of a statement is the least type it belongs to, i.e. the minimal expressivity needed to encode it. Examples of this are Logical Complexity (section 8.20 on page 146), and Kemeny's measures (section 8.18 on page 146).

The effect of this analysis can depend on how fine-grained the type classification is. If the difference between adjacently expressive types is large then this analysis will not specify much about the expression, as many different possible expressions will be possible within each classification, all differing in complexity, allowing the possibility that an expression of a higher type could be intuitively simpler than that of a lower. This is especially likely if the hierarchy of expressivity only encodes one aspect of the expression's potential complexity. For example comparing arithmetic expressions over the integers and reals, the solution set of  $x^2 = 2$  can only be expressed in the reals, but is much simpler in every other respect than many equations over the integers<sup>32</sup>, similarly the task of computing the number of zeros of an equation is computable over the reals but not over the integers [128].

If the analysis is fine-grained and fairly complete in its coverage of the potential aspects of complexity in an expression, then this comes close to a language of specification to which the complexity of the expression can be judged.

### 3.4.10 Dimension

The number of dimensions a model *requires* is some indication of its complexity. Simple systems can often be depicted with one or two dimensions, whilst if a model needs many dimensions, it might indicate that the relationship between these are necessarily

---

31.This is not always the case, for example compare the theory of real and integral arithmetic, many of the problems (e.g. uncomputability) disappear when you move to reals.

32.As in the equation constructed in Chaitin [102], where the solution set is, in a real sense, unformalisable.

complex, in that they can't be reduced to fewer. Cognitive complexity (section 8.5 on page 139) and the dimension of the attractor in chaotic systems (section 8.9 on page 141) use this approach.

The number of dimensions is essentially a limitation on the expressiveness of the possible formulations. This is thus really a sub-case of complexity as expressivity (section 3.4.9 on page 65).

Dimension is thus at best a coarse limitation on one aspect of a formulation's complexity. For example, to characterise the full range of cylinder shapes you need two dimensions: height and radius, but there may be no relation between the two; to characterise the cylinder shapes of fixed aspect ratio, such that their heights are prime numbers (in some unit of measurement) takes only one dimension.

#### **3.4.11 Ability to surprise**

It is difficult to model complex systems, so it is likely that any model we have is incomplete in some respect. If we have come to rely on this model (for instance when the system has conformed to the model for some time or under a variety of circumstances) and the system then deviates from that model, then we are surprised. The ability to surprise is not possessed by very simple and thus well-understood systems, and consequently comes to be seen as an essential property of complex systems.

This is not useful as a complete categorisation of complexity for several reasons. Firstly, it is only relative to the sophistication of the model and our reliance on it. Secondly such surprise could be the result of other things like simple ignorance or error in model formulation. For example, we do not know whether the processes behind some apparently unpredictable quantum effects are due to a complex internal chaotic process or a simple fundamentally random one. Chaos theory tells us that completely deterministic mechanisms can generate sequences of data that are statistically indistinguishable from random sequences. This poses the question of whether such "random" processes are, in fact, just very complex ones and hence merely reflections of our paucity in modelling (see also section 6.2 on page 126 and the section entitled Order and Disorder on page 203).

#### **3.4.12 Logical strength**

One can think of statements as "holding within themselves" their logical consequences, which are revealed when one applies the machinery of a logic's proof

theory to them. This picture is formalised when one identifies a formal proposition with its logical theory. Combining this picture of propositions with the principle that systems are at least as complex as their subsystems leads to the conclusion that logically stronger propositions are more complex<sup>33</sup>. Löfgren [293] uses logical strength as the basis for a measure of ‘interpretative complexity’.

Viewing propositions like this compels one to accept that either the logic has been effectively simplified by identifying all equivalent propositions with the same theory or that each theory has a complex (and often infinite) set of labels. Propositions would essentially be considered only from the point of view of their proof theory and not from other frames of reference (e.g. their syntactic structure). This is especially obvious when the contradiction  $p \wedge \neg p$  is considered, this is the strongest possible in classical propositional logic and so by this characterisation must be the *most* complex!

### 3.4.13 Irreducibility

Irreducibility is a source of complexity. A classic example is the three body problem in Newtonian mechanics, where the goal is to solve the equations of motion of three bodies that travel under mutual gravitational attraction. This is analytically unsolvable and hence is qualitatively different from any reduction to several separate 2-body problems<sup>34</sup>.

To characterise complexity as irreducibility is too extreme, for two reasons. Firstly, this would mean that many formal systems would be counted as simple (see the example in section 3.1.3 on page 44). Secondly this would rule out many meaningful comparisons of complexity, for example if comparing three systems (primordial mud, a simple organism, and us), one would be forced to categorize two of these as equal in complexity (this is also the case with the second example in the “Complexity as a comparison” section above (section 3.2 on page 46).

If irreducibility is to be allowed degrees and can be meaningfully defined so that formal systems do not necessarily come out as simple then these problems are mitigated.

For examples see section 8.17 on page 145.

---

33.Or, at least, not simpler.

34.The five-body problem is even worse: five bodies travelling at finite speed initially can interact so that they all disappear to infinity in a finite amount of time.

### 3.5 Complexity is Relative to the Frame of Reference

Repeatedly, in the analyses immediately above, we saw that the effective complexity depended on the framework chosen from which to view/model the system of study. This framework is very close to what Suppe called the ‘disciplinary matrix’:

*“The disciplinary matrix contains all those shared elements which make for relative fullness of professional work, models, ontological commitments, symbolic generalisations, a language, with meanings specific to that community, some interpretive symbolic generalisations, and so on.” [428] p.495*

In this subsection I will highlight some of the aspects of this framework in order to prepare the ground for the definition of complexity in the next chapter.

#### 3.5.1 The level of application

In the section on complexity as the midpoint between order and disorder (section 3.4.7 on page 62), we saw how the level of description greatly affected the interpretation of its complexity. When we looked at the patterns themselves we came to a different conclusion than when we were considering possible rule sets to generate those patterns. This is also inherent in many of the examples (e.g. the gas example in section 3.1.4 on page 44) or analyses above.

In fact, complexity often appears when we are seeking to cross levels. In the ant colony example (section 3.1.5 on page 45), I noted the great contrast between a macroscopic and a component view of the system; the complexity occurred when I sought to explain the former in terms of the latter. When we keep to a very similar or even unitary framework, systems are often simpler; when we sought to explain the behaviour of the ant colony in terms of both stimulus and response to the whole colony our task seemed a lot easier.

The criticality of scale in the modelling of phenomena leads Badii and Politi [36] to focus their characterisation of complexity solely on such hierarchical and scaling effects. As they say:

*“The study of the scaling behaviour of physical observables from finite-resolution measurements appears, therefore, as an essential instrument for the characterisation of complexity” [36] p.249*

### **3.5.2 Goals - type of difficulty**

In the gas example (section 3.1.4 on page 44), the complexity of the gas depended on whether we were trying to explain its behaviour statistically or deterministically (i.e. with or without “randomness”).

When considering processing time as a measure of complexity, (section 3.4.4 on page 59), in the checker-board colouring example the difficulty in terms of time was more fundamental than that of program space: the AIC (section 8.2 on page 136) of all the coloured checker-boards is small, but the computational complexity (section 4.3.3 on page 84) is large. In other examples, like that of producing a random sequence, the AIC will be large but the computational complexity small, illustrating how the complexity is relative to the task in hand, which is of course relative to your goals.

Another contrast is between the difficulty in trying to find (or induce) a suitable model description compatible with set of data and that of trying to analyse a given model description in terms of the properties of its content. Models that are simple to analyse can sometimes be very hard to find.

### **3.5.3 Atomic parts**

We saw how the base units of our descriptive framework affected the complexity of our models. This was illustrated by the gas example (section 3.1.4 on page 44) - whether we included “randomness” or “noise” as a basic (and thus unanalysed in this framework) element of our framework effected the complexity of the description.

Similarly in the ant colony example (section 3.1.5 on page 45), whether we took the individual ants as the atomic units of our description or just looked at the colony as a whole mattered greatly.

Which parts are considered “atomic” (or merely more basic or primitive) is just one aspect of the language of modelling or description, but it is important as it ‘anchors’ what one’s starting points are to be. For example, the complexity of determining the tape that results from the action of Turing machine may be very different from determining the Turing machine that will terminate with a given tape.

### **3.5.4 The language of description**

That the language of representation and modelling is critical to the effective complexity, is clearest in the programming language example (section 3.1.8 on page 46).

Here the language is completely explicit and makes a clear impact upon the task in hand. This example has many parallels with the mathematics example (section 3.1.3 on page 44). In mathematics there are two distinct difficulties: learning to use abstract and expressive mathematical languages and using these to solve a problem. Frequently if a problem appears insoluble the solution lies in shifting to a more powerful and expressive language to attack the problem from<sup>35</sup>. Toulmin goes further, he says:

*“The heart of all major discoveries in the physical sciences is the discovery of novel methods of representation, and so of fresh techniques by which inferences can be drawn – and draw in ways which fit the phenomena under investigation. The models we use in physical theories, . . . are of value to physicists primarily as ways of interpreting these inferring techniques, and so of putting flesh on the mathematical skeleton.” [438] p.34*

In the car engine example (section 3.1.6 on page 45) we saw how the different ways of modelling it effected not only our perception of its workings but also our method for interacting with it. These models depended crucially on the framework of mechanical understanding we had. If we had looked at a slightly different engine (e.g. a steam engine) we would form slightly different models of it, but these would very probably be on a par in terms of sophistication with those we formed of the car engine. The critical determinant of the complexity of our models arose not so much from the happenstance of the particular example as from our language of understanding and representation of engine mechanics.

I will, like others (e.g. Kauffman [249]), restrict my application of the concept of complexity to representations within specific language systems. Thus talking about complexity will necessitate indicating the language of representation that this is relative to. Here I intend the term “language” to have a wide interpretation which includes, but is not limited to, formal and natural languages. I am essentially using language as the most powerful idea available to capture the interaction and dependence of expressions of models to a framework.

---

35. See also Simon in [412].

The closeness of the relation between models and the language they are expressed in is implied in the following quote from Kuhn:

*“So if anyone asks: ‘What more is there to look at in science besides the models, the actual phenomena, and the relationships between them?’ we can answer ‘The structure of the language used in a context where a scientific theory has been accepted.’” [161] p.44*

## 4 A Definition of Complexity

*Complexity is that property of a model which makes it difficult to formulate its overall behaviour in a given language, even when given reasonably complete information about its atomic components and their inter-relations.*<sup>36</sup>

This is a general definition, which is intended to have different interpretations in different contexts. It relates the difficulty in formalisation of the whole to that of the formalisation of its parts (typically a ‘top-down’ model compared to a ‘bottom-up’ one) in the language. It is only applicable in cases where there is at least a possibility of gaining significant information about the components, thus clearly separating ignorance from complexity. Different conceptions of complexity depending on the base language chosen, the type of difficulty focused on and the type of formulation desired within that language.

The important aspects of this approach are that:

- it applies to models rather than natural systems;
- complexity is distinguished from ignorance;
- it is relative to the modelling language it is expressed in;
- it relative to the identification of components and overall behaviour;
- complexity is a global characteristic of a model;
- you will get different kinds of complexities from different types of difficulty;
- complexity represents the gap between component knowledge and knowledge of global (or emergent) behaviour;
- since difficulty is sometimes comparative, complexity will be also.

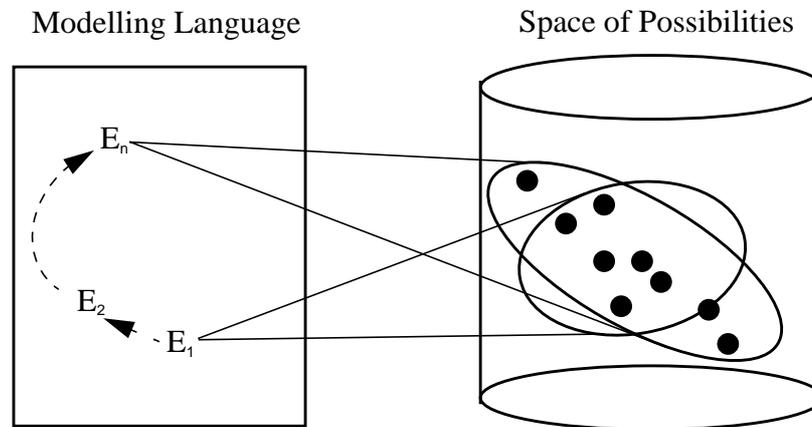
Using the picture of modelling I presented in section 2.3 on page 33 complexity is the difficulty of finding a description of the overall behaviour of a model. This description is also a model description, so we are dealing with (at least) two models here: the model of the ‘components’ and their interactions and the model of the resultant overall behaviour. I will call the former the *object model* and the later the *overall model*. The complexity is

---

36. Some aspects of this have been pointed out independently by other authors, e.g. Grassberger [194] talks about complexity as “difficulty”, Kaufman [249] relativises it to the language of representation, Raccoon [365] talks about comparing a top-down vs. bottom up approach, etc. but I have not see these aspects combined in this way.

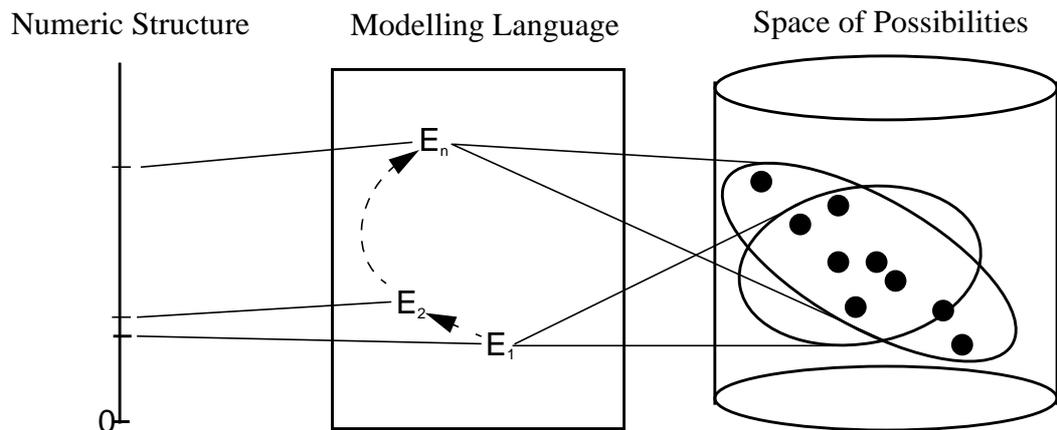
thus the difficulty in finding an overall model description for the resultant behaviour of the object model.

Perhaps the simplest case is where the object model is a data model and the goal is to find a description in some modelling language whose content subsumes the data model content and is as specific as possible. The difficulty of ‘finding’ this description depends upon how this is sought and the criteria for what would constitute an acceptable solution. This is the basic situation that occurs in many machine learning set-ups (as discussed in section 2.4 on page 38), where the problem is to design or discover process to find such descriptions. The situation is illustrated in figure 17 on page 73.



**Figure 17.** Finding a description to cover a data model

Then *measuring* this difficulty implies a mapping onto a numeric structure. The ‘search’ process occurs by acting on the model descriptions so that the measure of difficulty is a measure upon the modelling language. Thus we have a ‘chaining’ of models (as described in section 2.2 on page 28), Thus the complexity here is a numeric model of the difficulty of finding a model description that meets some criteria for matching (e.g. by what I called subsumption or approximation in section 2.2 on page 22) the data model of the phenomena under consideration.



**Figure 18.** Measuring the difficulty of finding a model description

Of course, most modelling situations are more complicated than this. In general the object model will not be a simple data model but one that supports some inferences in the form of predictions from initial conditions. The method of finding a description for the resulting predictions might not be an inductive one but an analytic one based on known properties of the description. The criteria for the sought-for formulation of overall behaviour might include the acceptability of trade-offs between the error, specificity and cost of finding this description.

There are many different types of complexity involved in actual examples of scientific modelling. Conflation of these into a single “complexity” of scientifically modelling a certain system will generally result in confusion.

There might be:

- the complexity of the data: the difficulty of encoding of a data model compactly given a coding language;
- the complexity of the informal (mental) model: the difficulty in making an informal prediction from the model given hypothetical conditions;
- the complexity of using the formal model to predict aspects of the system under study given some conditions;
- the complexity of using the formal model to explain aspects of the system under study given some conditions.

Each of these will be relative to the framework it is being considered in (although this and the type of difficulty may be implicit). Further if attention is switched to the

process of measurement involved in the production of the data or the process of improving the model then you get more corresponding complexities. Some sets of complexities will be easier to relate than others: there may be a close link between the complexity of the data and the complexity of the formal model to explain that data (relative to an encoding of that data into the appropriate formal language) but a more distant link between the complexity of the data relative to a formal language and the informal model used by the scientist in order to guide her search.

To take a more concrete biological example, the biological complexity of a model of an organism (the difficulty of explaining its functioning in terms of its genetic make-up) and a sort of environmental evolutionary complexity (the complexity of a model of its behaviour with respect to an environment in terms of the evolutionary pressures that created it) may be almost completely unrelated. Unfortunately, such different complexities are often conflated into one. This has considerably muddied the debate about whether there has been an increase in the complexity of organisms over the course of evolution (see [71, 221, 147] and section 6.6 on page 130).

In general, the complexity of natural systems or natural phenomena are closely associated with the complexity of their most appropriate models, but typically, in order to objectify the derived measure of complexity these models are relativised to some privileged framework (where this privilege is either argued for or established by the traditions and methodologies of the field). Which are the most suitable choice of models can be greatly influenced by the modelling language and the tolerance of error and vagueness. For more detail on this see Appendix 6 - Complexity and Scientific Modelling on page 199.

There are a number of aspects of this definition that need clarification. I will discuss these points below.

## **4.1 Aspects of the Definition**

### **4.1.1 Identity of a system**

Identifying *whole* systems is problematic in general. Many natural systems are causally interconnected, making any division seem somewhat arbitrary from anything approaching a general viewpoint. This problem is not solved by the seeming ease with which we actually *do* assign useful identities to systems as this seems to result at least

partially from a blend of psychological, pragmatic and accidental features. There may well be true philosophical aspects of this distinction and I will briefly discuss how ideas of complexity may be a part of this account (section 6.8 on page 132), but since we are restricting ourselves to particular representations of any system in a language this is not a problem for us; we *decide* the level of interaction of the system with the rest of the representation when we specify our representation and *assign* an identity to some portion thus making it our focus. This does not eliminate the problem, of course, as it then becomes critical *how* we model any real system if we want to discuss the complexity of an aspect of it, but such concerns are common to many aspects of using representations of real systems.

So for us a *system* may just be a collection of statements intended to describe it sufficiently. We necessarily decide questions of *system identity* as a by-product of representing the system to start with.

#### **4.1.2 Atomic components**

An atomic component is one that can not be reduced in terms of other components in the representation chosen. The above definition thus only refers to systems where a base layer of atomic components can be identified, because it is a definition that compares descriptions. The choice of modelling framework will include deciding what constitutes the components and the overall behaviour (see section 4.1.4 on page 78).

Not all representations will have atomic components. For example, one might consider a web of social cause and effect. Here every cause might itself have several sub-causes and every effect be composed of sub-effects. When tracing this web downwards one may not reach any ultimate cause. The representation of this might well be recursive, if not circular. However, if this is to be formulated in terms of a modelling language, there will have to be a limit on the number of symbols used. This limited model can be considered as an approximation of the representation.

Taking such a series of ever more detailed models might result in a series whose complexity grows unboundedly. In this case one might be justified in describing the complexity of the target system as infinite. On the other hand one might come to a definite limit, where increasing detail increased the complexity progressively less, indicating a limit, but this limit might not be unique. If one took a different series of system approximations one might well get a different perspective and limit. In this case one could

not ascribe a single complexity to the system with respect to this framework, unless one reduced the scope of the study to a few aspects to ensure uniqueness (or at least close convergence). These different series of approximations could be viewed as different perspectives each giving a different complexity or else you could view it as simply undefined since it would not have identifiable atomic parts in this sense.

In other words, if the approximations do not converge to the intended representation then one would have a (greater than usual) problem with ascribing meaning to them in terms of the target system anyway. This problem with ascribing meaning would include the sense of the complexity of these approximations. To resolve this problem one would need to change or augment your representations in order to make their intention reliably determinable, one aspect of which is how they approximate to your intended model. Then, if this method of approximation also allows one to approximate a meaningful idea of complexity then this complexity can be taken as the limit of the complexities in the approximations. Otherwise it is doubtful whether such an identification could be made sufficiently stable to be at all useful<sup>37</sup>.

Another possibility is that, although there are no “atomic” parts at one level, there is a meta-level description where there are such. For example a directed digraph may have no unique starting node, but there might be a description of it in terms of nodes and arcs where the symbols representing these might be basic.

### **4.1.3 Difficulty**

There are many possible kinds of difficulties in formulating the overall behaviour of a system. Each will have a correspondingly appropriate measure of complexity (see section 4.3 on page 83 for examples of this).

If the size of the representation was the primary concern, i.e. the memory requirements for the storage of it, then this criteria of size represents your measure of complexity (see section 3.4.1 on page 57. For example, this might be particularly relevant for humans when they have are dealing with a fairly meaningless representation (like numbers) when they had no memory aids such as pencil and paper. The longer it is the

---

37. Several approaches in physics aim to explicitly capture this scaling behaviour, see Appendix 6 - Complexity and Scientific Modelling on page 199 and section 8.16 on page 145 on measures to capture hierarchical scaling behaviour.

more difficult it is to deal with. This difficulty is easily overcome with simple aids. It is also very sensitive to the language of representation. It is notable that humans tend to choose internal representations that circumnavigate this limitation in short-term memory or “chunking” as it is sometimes called<sup>38</sup>.

If one has plenty of memory, *one knows the algorithm* and is in a situation where one has to deal with example problems of indefinite size, then the time the overall behaviour takes to compute compared to the problem size will present the major difficulty. This produces the time measure of computational complexity (see section 8.47 on page 162) and their corresponding classes (P, NP, NP complete etc.). This represents more of a difficulty than mere space problems, but it assumes that the problem of *finding an algorithm* has already been solved.

From a philosophical viewpoint the difficulty in *reducing* a big problem to ones involving more fundamental units is a more major problem. This is the basic analytic problem. There are different ways of tackling this, many of which involve changing the language of representation (see section 5 on page 86). It is this aspect that may be most important when studying the complexity of formal systems.

#### 4.1.4 Formulating overall behaviour

What constitutes the *global* behaviour of a system must be, to some extent, decided by the observer. For example, the overall properties of propositions may be considered from the point of view of a proof theory of a logical language with respect to a language of its theories. In less restricted systems, like a system of differential equations, there may be a range of global behaviours to choose between.

This can not be an arbitrary choice, however. There are limitations to what could be considered a global description. I list some below.

- The properties of the parts must determine the overall behaviour to a large extent. This expresses something of the essential relation of parts and wholes. If they do not determine it, then the “complexity” of formulating the behaviour relative to these parts would be fairly meaningless.
- More general solutions should be preferred to mapping more specific properties of the system. A general analytic solution to a system of differential equations is

---

38.From Miller [325]

to be preferred over its value in a specific range.

- The language of representation can limit the choice of overall behaviour available for formulation.

These constraints still leave considerable choice as to what could be considered as “overall behaviour” and what the atomic components might be. In many cases it might be arbitrary which is which, for example looking at a language’s syntax and semantics - you might consider the complexity the difficulty of determining the syntax from the semantics or vice versa,

Often the question of what it means to formulate the overall behaviour is closely linked to the concept of “difficulty” considered relevant, see section 4.1.3 on page 77 above. For example, if “difficulty” means the computational time taken to compute the result of a program, then the formulation of the overall behaviour is the results or trace of that computation.

#### **4.1.5 Complexity vs. ignorance**

The most basic distinction that this definition makes is between difficulty due to ignorance and difficulty due to complexity. You can only reliably attribute complexity to a system when there is a possibility of knowing a reasonable amount about its components, otherwise the apparent difficulty of formulation might be merely due to some simple but unknown mechanism. Specifying *complete* knowledge of the components and their interactions is frequently impractical and often unnecessary; you sometimes can have sufficient information to rule out any simple unknown factors but still be faced with complex global behaviour.

For example an element of randomness is frequently taken to be atomically simple in languages of representation because there is no possibility of obtaining information about its general process (e.g. error terms).

One corollary is in situations where a large amount of ignorance is inevitable (e.g. the universe at the instant of the big-bang). The above definition will just not apply to such situations as near complete information about their components will not be available. Whether this means that complexity is *inapplicable* to such cases or that we will just *never know* the extent of their complexity does not make any practical difference to the use of the concept. This is the core of the arguments in the previous section see section 3.3 on page 47 for more on the scope of the idea of complexity.

#### **4.1.6 As a gap between the global and local**

Often complexity represents a gap between the ease of representation of the component parts and the difficulty with respect to the overall behaviour of the system<sup>39</sup>. This is a key characteristic of *emergent* behaviour.

Thus a model of the brain may be seen as complex because of the difficulty of explaining its over-all behaviour from the interaction of its parts (usually considered to be its neurones). The extent of the irreducibility of this difficulty is due to the fact that we are largely ignorant of the parts' processes and interactions.

This is not to say that all complexity must result in a contrast of levels of description. It is just when the contrast between different descriptions is so marked that it is practically impossible to translate between descriptions (great complexity) that this produces emergence. An example is the gap between the mental and physical worlds [116]<sup>40</sup>.

#### **4.1.7 The comparative nature of complexity**

Since, given a similar framework, difficulty is comparative so is complexity. Much of the utility of a concept like complexity comes from estimating the degree of difficulty of a problem and in attempts at simplification. In order to be able to use these aspects the model of complexity chosen needs to be comparative; you need to be able to say that one representation is more complex than another. For example, you might wish to compare the complexity of different axiomatisations of a logic.

The extreme case of merely classifying systems as either complex or simple is a (sometimes less useful) special case of this (see section 6.7 on page 131). It is merely a much coarser ordering. There are those who reserve use the label of "complexity" for holistic systems<sup>41</sup>. This is fine but tends to deprive the word of many of its connotations. This is often a reaction to limited internal system-theoretic models of complexity which take one aspect of complexity with respect to only one framework for which some sort of privilege is claimed<sup>42</sup>.

---

39. See Raccoon in [365] for a clear account of this.

40. Pribram in [360] even attributes some complexity to the 'gap' between the hemispheres of the brain!

41. Or non-mechanistic systems, see Rosen [389]

#### **4.1.8 The existence of complexity**

It may be argued (by an extreme optimist) that *all* difficulty in formulation of overall behaviour is due to ignorance (as Waxman did in [463]). In which case complex systems would not “really” occur, i.e. all complexity is only apparent. Counter-examples of a formal nature tend to discount this possibility (see the example in section 3.4.5 on page 60 above).

Another more common example is the game of chess. Here the component rules and their interaction are completely specified, but the questions of overall specification, like determining a perfect strategy are still very difficult.

#### **4.1.9 Relativisation to a language**

Many examples from the previous chapter (section 3 on page 44) illustrate how the complexity of a representation can change when the language it is expressed in changes. This is a vital property of complexity as one of the principal tactics (as far as humans are concerned) used to simplify representations is to change the language they are embedded in (section 5.7.4 on page 123).

I mean the scope of “language” to be fairly wide - it will include informal as well as formal languages, natural as well as symbolic. Of course, the more informal the language, the more informal the derived measure of complexity it will imply relative to some subset and difficulty. It is doubtful you could formally show a complexity comparison between two poems, unless the scope of the language they were compared in and related to was sufficiently specified.

## **4.2 Examples**

In addition to the examples of common usage above in section 3.1 on page 44, I present a few examples to illustrate the workings of the above definition. As I will deal with formal languages below (section 5 on page 86) these will be concerned with other areas.

---

42. It is ironic that it is the holists who have reduced the meaning of complexity and simplicity to that of natural and formal systems.

#### **4.2.1 The flight behaviour of a herd**

Consider a hypothetical herd of animals, in which the flight behaviour of each individual is fairly well understood in terms of its reaction to danger, its wish to follow others of its kind, etc., where we are trying to understand the flight behaviour of the whole herd in terms of its overall direction and path. If it turned out that all the animals always followed one specified leader, or all went in a direction represented by the average of the separate directions they would have gone individually then we would be justified in calling the flight behaviour of the herd simple. If the behaviour turned on the precise configuration of the herd at the time of attack, so that the animals followed different individuals at different instances as the configuration developed, we would be justified in calling the behaviour more complex.

To illustrate how the language can make a difference consider a case where the language did not take into account the direction of travel (of individual or herd) but only the distance and average speed. In this case, the behaviour might be simple, despite the fact that the behaviour is complex if you had to take the direction into account.

The type of “difficulty” can also be crucial. Given a language which includes direction and speed, a predator might be concerned with the difficulty of predicting in which direction the herd will initially set off, while an ecologist might only be concerned with the eventual direction and distance a herd travelled until it settled down. The first might be difficult to predict, the second easy; in the first case the behaviour of the herd would be complex while in the second it would be simple.

#### **4.2.2 Cellular automata**

Cellular automata have become icons of complexity in physics (due to Wolfram’s promotion of them as such [471]). They are described in very simple local terms that are easy to compute (simple enough that one could imagine them modelling real physical properties) but have complex global properties<sup>43</sup>.

Here the specification of the parts is easy (an initial binary pattern and a simple rule for determining each bits next state, e.g. for each bit: 1 if the sum of bits of it and its two neighbours is odd, 0 otherwise). The complexity of the result after a certain number of iterations given the initial rule and bit pattern is dependent on the language of such

---

43.For an account of this see Wolfram [473].

formulation. It may be easy to formulate some statistical results for a large class of such rules but very difficult to predict if certain sub-patters will occur.

### **4.3 Relationship to Some Other Formulations**

Broadly, existing measures of complexity can be seen as either:

1. *A special case of the approach given above in Section 4 above;*
2. *A relativisation of the approach either to some physical attribute (e.g. scale) or some 'privileged' framework to 'objectivise' it.*
3. *A weak characterisation of complexity better suited to some other descriptive label, such as 'information'.*

I consider a few existing formulations of complexity below, for some others see Appendix 1 - A Brief Overview of Some Existing Formulations of Complexity. The strength (or otherwise) of their claims as important or general measures of complexity thus rests with the critically of the type of difficulty they measure, the range of systems they apply to and the appropriateness of the underlying descriptive language.

#### **4.3.1 Number of inequivalent descriptions**

Extending the approaches to complexity taken by Rosen and Pattee (see section 6.7 on page 131), Casti [88] defines complexity as the number of nonequivalent descriptions that an observer can generate for a system it interacts with (see also section 8.29 on page 152). The observer must choose a family of descriptions of the system and an equivalence relation on them - the complexity is then the number of equivalence classes the family breaks down into given the equivalence relation.

This can be seen as a special case of my definition where the language of modelling is implicitly determined by the family of descriptions, their equivalence relation, and the relevant *difficulty* (being that of the coherence of the descriptions). If one had only one possible description that was completely coherent, one could claim that (with respect to the language) one had the representation of the system. If there were more then there were evidently incoherent aspects in the description that admit to alternative but not simultaneous description. See section 8.29 on page 152 for examples.

The trouble with this is that it is very difficult to ascribe any coherent meaning to the resulting measure, especially if one goes beyond a categorisation of one/many/infinite. In many languages (especially infinite ones) once one has two inequivalent descriptions of something then one can immediately construct a third inequivalent description by mixing parts of the first two, then construct a fourth and so on indefinitely. In other, suitably expressive languages one can always construct inequivalent descriptions given *any* particular subject and non-trivial equivalence relation, in which case there would be no simple systems. This makes using this particular measure of complexity difficult to use to make fine complexity judgements.

#### **4.3.2 Effective measure complexity**

EMC [194] can be seen as the difficulty of predicting the future values of a stationary series, as measured by the size of regular expression of the required model. See section 8.16 on page 145 for other related examples of this kind of approach.

#### **4.3.3 Computational complexity**

Computational complexity (section 8.47 on page 162) can be seen as the asymptotic difficulty (in terms of computational resources - typically time or storage space) in computing the output relative to the size of the input, given the specification of the problem. Here the difficulty is that of time. The overall description is the results of the computation, the bottom-up description is that of the problem, where the system is the computer program. Again this is a special case of my definition when one already has the ideal program for calculating the results. It is thus a residual complexity concerned with the practicalities of computation once one has solved the analytic aspects.

#### **4.3.4 Algorithmic information complexity**

Again, this can be seen as a special case. The algorithmic complexity is the minimal difficulty of storing a program to reproduce any particular pattern/number/index<sup>44</sup>. The overall description is the program and the bottom-up description is the original pattern and the language of description is (typically) a Turing machine. This is thus an indication of

---

44. Alternatively this could be taken as an estimation of the time taken to find an expression in a bottom-up enumerative search.

the amount of essential information (i.e. incompressible with respect to a Turing machine) inherent in the pattern. For more on this see section 8.2 on page 136.

As argued above (section 3.4.1 on page 57) information is at best a weak measure of complexity unless applied at an abstract level.

#### **4.3.5 Shannon entropy**

Shannon Entropy, can be seen as the difficulty of guessing a message passing down a channel given the range of possible messages. The idea is that the more difficult it is to guess, the more information a message gives you. This was not intended as a measure of complexity, but has been used as such by subsequent authors. See section 8.36 on page 156 for related approaches.

#### **4.3.6 Crutchfield's "topological complexity"**

The number of states in the smallest model in the minimal language where this is finite, is called the "topological complexity" in [122]. This is the difficulty of finding (in a bottom-up search procedure) a correct model, once the appropriate language has been chosen. Here the notion of complexity is relativised to the hierarchy of formal languages. This is not necessarily uniquely defined as the formal languages are only in a partial order. Also when it comes to stochastic languages, a minimal model roughly corresponding to "anything can happen" could generate any data. See section 8.32 on page 153 for related approaches.

## 5 Applications of Complexity to Formal Languages

### 5.1 Types of Complexity Involving Formal Languages

As noted in the previous section, complexity is relative to the kind of difficulty that concerns one most. Frequently this is the limiting factor in a task.

For example, if one has already programmed a computer for a certain search task, the remaining difficulty is represented by the time and memory this search will take to complete. If, further, one is in the happy circumstance where the speed and capacity of computers is growing exponentially (as it has been in the last few decades), then a critical factor will be whether the time and memory requirements also grow exponentially with the size of problem, or merely in a polynomial fashion. If the later is true, one will only have to wait, for the problem to become tractable (assuming the current trend in computing power continues), while this is not the case, the program is likely to remain intractable using this algorithm. This sort of question is dealt with by the measures of Computational Complexity (see section 8.47 on page 162).

Similar examples of possible limiting factors include the size of matrix required for discriminating the independence of expressions (section 8.40 on page 158), the ease with which expressions can be compressed (section 8.2 on page 136), the length of proofs (section 8.19 on page 146), the predictive difficulty of modelling using regular languages and a language's position in the Chomsky hierarchy [122].

One I wish to concentrate on is that corresponding to the difficulty in analysing expressions in a formal language, which I will call the “analytic complexity” of an expression. In order to analyse an expression, one tries to trace its roots, i.e. to decompose it according to some system. The language within which one attempts this will be critical. For example some logical theorems may be quite simple to decompose as an expression but difficult to decompose in terms of proof, for other theorems the opposite might be the case.

Such an analytic process can be seen as a possible strategy for *deducing* the overall properties of a model or expression. In this light, analysis by decomposition is complementary to *inducing* the properties by seeking a model for there properties. In other words the ease with which a model can be decomposed reflects that model's accessibility to analysis and in this way obviates the need for a lengthy inductive search process. So in

situations where analysis is feasible it becomes the critical factor in the search for an overall formulation of its behaviour.

## **5.2 Expected Properties of “Analytic Complexity”**

Firstly I will look at what properties one might expect of such an “analytic complexity” in the context of formal languages.

### **5.2.1 Independent of the particular symbols used**

A measure of complexity on statements in a formal language should be independent of the particular symbols used; it is the pattern of the expression that encodes its meaning. It should be immaterial whether you write “ $a \wedge b$ ” or “statement1 and statement2” if the corresponding formal languages are isomorphic.

### **5.2.2 The complexity of sub-expressions should be less than the whole**

That a sub-system or sub-expression should be no more complex than the system or expression it is part of, is perhaps the most commonly accepted property of complexity. It is rare that a proposed measure of complexity does not obey this rule (but Abstract Computational Complexity is a rare counter-example, section 8.1 on page 136).

If one considered that, in some circumstance, an expression was less complex than one of its parts then, presumably, there would be some mechanism for preventing this “buried” complexity from being fully realised in the whole. In this case, only an abstraction of the full expression is being considered, in which case the complete representation of this abstraction would be different from the complete expression. Sometimes this is because of a confusion between the syntactic content of an expression and some other aspect.

So, for example, in a first-order logic with equality one might consider all substitution instances of identity equally simple because from a proof theoretic view they perform a similar role, despite the fact that some of these instances seem to (syntactically) contain arbitrarily complex sub-expressions. Either for their purposes they are considering all such instances as the *same* as identity, in which case a different language (where this identity is enshrined) is essentially being used, or they are considering the expression from within the language of its proof theory. In either case the language used is not the syntax of the expressions (over which the sub-expressions are being taken) but the syntax of some

proof theory, so it is not surprising that it does not have these complex parts when expressed in these ways.

### 5.2.3 Expressions with no repetitions are simple

Since I am, for the moment, considering complexity over a potentially wide range of languages I need to define what I mean by atomic statements. By atomic statements I mean expressions that have no constituent syntactic parts (in that language). Of course it is possible that in some systems there are no atomic statements (see section 4.1.2 on page 76).

Atomic expressions are as *syntactically* simple as you can get. It is reasonable to allocate them a minimal complexity. Similarly, in a proof theory, a one line proof just listing an axiom would be a minimally complex proof.

One might object that syntactically atomic symbols may have complex meanings, but if we were to attempt to *measure* the semantic complexity of models, we would need to formalise this in a language so as to distinguish the relevant semantic properties. The “semantic complexity” would be relative to this new language.

Expressions with unrepeated atomic symbols, do not *use* the reference denoted by such a symbol, because for there to be any meaning one needs somehow to establish an identity indicated by a symbol and then make some statement about it. If the only reference by a symbol is made once then that reference specifies its sole property (for the expression), there can not be anything more. It is like answering the question “Who wrote the Iliad?” with the name “Homer”, but if the *sole* property of the name “Homer” is that it is the author of the Iliad then it does not get you very far. If we knew anything else about Homer (for instance that he was a single historical being<sup>45</sup>), then the appellation could be meaningfully used, and some complexity appear.

Of course, if one is talking about a language which relates *groups* of sub-expressions, then the symbol may be meaningfully repeated elsewhere (in another sub-expression) to establish its purpose. Logical constants in various proof theories have this role. Again, it is necessary carefully to distinguish between the complexity of expressions within a syntax and the complexity of proofs within the syntax of a proof theory.

---

45.I am told that this is unlikely.

For example, an expression with no repetitions, like “When at a constant pressure, heat causes gases to expand.”, may have a complexity within a language of scientific verification, as the definitions of the words are all established elsewhere. If, on the other hand, this sentence was used where none of the words had a definition or defining context, it could, on its own, perform no complex role.

#### **5.2.4 Small size should limit the possible complexity**

In *very* small systems, there might be a limited number of different relationships between its components. In this case, one would expect that there would be a limit to its complexity. However, the complexity of a system can grow extremely quickly with its size, for example there is a Turing machine with only five internal states that halts after exactly 23,554,768 steps [303].

#### **5.2.5 There should be no upper limit to complexity if the language is suitably generative**

In most situations one feels that there is a limit to how far one can simplify statements, but it is always possible to complicate them. The whole point of a generative language is that one can construct an expression as complex as necessary. So it would be very surprising if there was an general upper complexity limit for expressions in such a language.

If there were such a limit, what could it mean? One possibility is that although the expressions may appear more complicated they were not “in fact” so, but in this case we are no longer dealing with the expression itself but that expression from within a different language context represented by the meaning of “in fact”. For example, there is an upper limit to how Algorithmically Complex (section 8.2 on page 136) you can prove strings to be from *within* a wide range of formal systems but this does not mean there was a limit to the Algorithmic Complexity of strings. On the contrary, most strings have an Algorithmic Complexity close to their length; it is a limitation on the power of proofs within the systems that is at the root of this limit.

Another possibility could arise if the expression's syntax were limited in some way so as impose an upper limit on its possible complexity. If such a language was generative it would have to generate an infinite number of expressions of limited complexity. Thus most of these expressions would have to not get more complex as they increased in size.

This would make for an odd language indeed, stuffed with unwieldy but simple expressions. They do exist. An example is a language with an infinite supply of atomic symbols and a connective “ $\rightarrow$ ”, where only atomic symbols and only expressions of the form  $x \rightarrow y$  are allowed where  $x$  and  $y$  share no atomic symbols. In most general expressive languages I would not expect this to be the case.

### 5.2.6 The complexity of irrelevant substitutions

One way of dealing with complex systems is to try and break them down into a number of simpler systems. If they are not broken down into *independent* sub-systems, there is a risk of losing some aspects of the original system in the process. If all the subsystems are mutually irrelevant to each other, then nothing will have been lost in the analysis. In this case the complexity of the whole system would be wholly in these sub-systems, for there is no other interaction between them at the system level.

In particular, if a sub-expression is substituted for an atomic symbol into another expression and this sub-expression is irrelevant to rest of the main expression, then the resulting expression could be completely and successfully analysed into those two parts. Thus we would think of  $(a \vee \neg b) \rightarrow (a \vee \neg b)$  as a substitution instance of identity  $x \rightarrow x$  with  $(a \vee \neg b)$  substituted for  $x$ ; in some systems the two levels would not syntactically interact. In the example the complexity of  $(a \vee \neg b) \rightarrow (a \vee \neg b)$  derives completely from the complexity of  $x \rightarrow x$  and the complexity of  $(a \vee \neg b)$ .

Here we would have to be careful to separate out the complexity of the expression's syntax from the complexity of its proof theory. If, in the logic's proof theory,  $(a \vee \neg b)$  and  $b \rightarrow a$  were inter-derivable and interchangeable then there would be some interaction between the levels in the sense that it would mean that an implication could imply another implication (itself).

There might well be a close connection between the complexity of an expression's syntax and its derivation (or theory) in a particular logic, but this is not necessarily the case. For example, in some inconsistent systems of Logic, the atomic proposition,  $a$ , might be derivable after a lot of work and so it might not have a simple derivation.

Another example is a system of differential equations. Some such systems are linearly separable, in which case the solutions can found separately for distinct variables. The variables do not interact, except that they both occur in this set of equations. You can

meaningfully talk about each separately without going into the details of their solutions and add the complexity back again by substituting the solutions later, without invalidating any of the previous discussion.

### 5.2.7 The complexity of relevant relating of expressions

When two relevant expressions are related then this relation is more complex than either of the parts. The sub-expressions  $A$  and  $(B \rightarrow B) \rightarrow A$  are relevant to each other. The joined expression  $((B \rightarrow B) \rightarrow A) \rightarrow A$  is clearly at least as complex as either of the sub-expressions it contains (by the sub-system property) and the top-level implication could add to this complexity.

In a densely connected system, where every part is relevant to every other, it is difficult to reduce the system to simpler subsets of the original, without losing important information. A popular way of expressing this is by saying that the whole is greater than the sum of its parts.

Imagine a very incestuous party, where everybody has known everybody else for a very long time. Here it is not possible to gain a complete picture of the system of relationships by studying sub-groups. Every time study is restricted to a subgroup of the total information is lost about some of the relationships which effect those in the subgroup. Conversely recombining these subgroups into the whole again is accompanied by a regaining of this lost complexity.

The simplest party is one where nobody interacts with anybody else. Such parties are no fun at all; nothing can come out of them. Everybody understands exactly what is going on.

A normal party is amenable to *some* analysis into expressions for sub-groups, classes, etc. but there are usually enough cross connections to rule out any *complete* understanding of the situation. We are thus sometimes surprised by their outcome (maybe this is why we have them).

### 5.2.8 Decomposability of expressions

Many of the above criteria are concerned with the decomposability of formula. Thus we will see that such decomposability is strongly related to analytic simplicity, as the easier it is to decompose without loss, the easier it is to analyse.

A statement in a formal language will be called decomposable if some coherent part of it (larger than a single sign) can be substituted for a symbol so that that part is irrelevant to the resulting statement. Thus the decomposability of a statement will be dependant on the syntax of the language and the definition of irrelevance (or relevance). The idea is that if such a substitution is possible then the two parts of the statement can be considered separately, as their only connection is though the symbol that now stands for the substituted part.

Relevance too will be defined relative to the syntax we are considering. I will take two statements to be irrelevant to each other if they do not contain any of the same sub-formula (in that syntax or in sub-syntaxes). This is a conservative definition of irrelevance; no doubt many statements (or examples of reasoning) that do share some of the same sub-structures could also be considered irrelevant to each other.

Thus a formula that can be repeatedly decomposed as described above will be considered simple compared to one that can not. This reflects our intuitive feeling of the complexity of such formula.

For example, the formula  $((a \rightarrow b) \rightarrow c) \rightarrow ((a \rightarrow b) \rightarrow c)$  is considered as identity  $x \rightarrow x$  with the sub-formula  $(a \rightarrow b) \rightarrow c$  substituted for  $x$ . On the other hand the formula  $((a \rightarrow b) \rightarrow c) \rightarrow ((b \rightarrow c) \rightarrow a)$  can not be simplified in the same manner.

I have argued that an number of constraints should apply to the complexity of formulas: a formula with no repetition of its sub-formulas is ultimately simple (I will refer to such formulas as “simple”) - it can be completely decomposed into single symbols; if a formula can be irrelevantly decomposed into parts then its complexity is just the sum of these parts; and that a formula must necessarily be at least as complex as any of its sub-formulas.

There will be formulas that can not be decomposed at all in this manner (in the same syntax). I will call such formulas “complex”. Thus I still have the question of how I compare such complexes with respect to their complexity. One way would be to call all such complexes equally complex (producing something akin to a discrete metric space), but this would not reflect our intuitions very well; identity would be as complex as sufficing!

### 5.3 Measures of Analytic Complexity

The above properties (listed in Section 5.2 immediately above), provide constraints on possible numeric measures of complexity. The characterisation of measures which meet these constraints and some simple results are now considered. I am not concerned with finding a set of axioms that show the existence and uniqueness of a measure but rather the converse – given the above properties I wish to find what measures exist which satisfy them. This reflects my stance that complexity orderings are not innate but are rather models of aspects of our descriptions to reflect the difficulty we have in finding or analysing them.

A practical result of this is that often the assignment of numbers to reflect the complexity comes before a complexity ordering. Occasionally these assignments are intended to give an indication of the exact level of difficulty (as in storage space), sometimes they are intended to merely give an indication of the order of magnitude of the difficulty, but often they are only intended to induce a complexity ordering of expressions. Thus the measures fall somewhere between the ‘ratio’ and ‘ordinal’ measures as categorised by Stevens in [423].

A decision to model the difficulty of finding or analysing a model description by complexity *measure* (i.e. a homomorphism into the reals) relies on the implicit assumption that the domain is sufficiently constrained so that it is meaningful to assume that a complexity ordering relation will be connected (that is for every pair of items  $x, y$  either  $x \leq y$  or  $y \leq x$ ). Usually this is entailed by the numerical definition of complexity on the whole domain of possible expressions.

#### 5.3.1 Notation

Let  $X \equiv_{\text{df}} X_0 \cup X_1 \cup X_2$ , be a set of symbols, where  $X_0$ ,  $X_1$  and  $X_2$  are disjoint sets. It is intended that  $X_0$  be a set of constants and variables,  $X_1$  be a set of unary symbols and  $X_2$  be a set of binary connectives<sup>46</sup>.

---

46. Although I only consider up to binary symbols here, the results could easily be extended.

Let  $L$  be a context-free language with the obvious production rules:

$$c \in X_0 \Rightarrow c \in L$$

$$u \in X_1, x \in L \Rightarrow (ux) \in L$$

$$b \in X_2, x, y \in L \Rightarrow (bxy) \in L.$$

The maximum depth of a formula,  $\text{depth}(x)$ , is defined recursively in the normal way:

$$x \in X_0 \Rightarrow \text{depth}(x) = 0$$

$$\text{depth}(ux) = 1 + \text{depth}(x)$$

$$\text{depth}(bxy) = 1 + \max\{\text{depth}(x), \text{depth}(y)\}$$

Similarly with the size of a formula,  $|x|$ :

$$x \in X_0 \Rightarrow |x| = 1$$

$$|ux| = |x| + 1$$

$$|bxy| = |x| + |y| + 1$$

Let  $\wp(x)$  be the set of well formed subformula of  $x$ . Note that  $x \in \wp(x)$ .

Thus  $x$  is a subformula of  $y$  iff  $x \in \wp(y)$ . This is a reflexive and transitive relation.

Define  $R(x, y)$  to be a minimal syntactic relevance relation on  $L$  thus:

$$R(x, y) \equiv_{\text{df}} \wp(x) \cap \wp(y) \neq \emptyset, \quad (\text{Defn R})$$

i.e.  $x$  and  $y$  share a common subformula. This is equivalent to sharing a common variable or constant. This is a symmetric and reflexive relation.

I will assume in all below that there is a sufficient supply of symbols in  $X_0$ , so that one can always find a symbol that is irrelevant to any particular formula. Formally:

$$x \in L \Rightarrow \exists c \in X_0, \neg R(x, c). \quad (\text{Suff Symbols})$$

If there is not add a suitable sequence of irrelevant symbols to  $L$ <sup>47</sup>.

Let  $x^{y/z}$  (or sometimes  $x(y/z)$ ) be the notation for the formula  $x$  where every instance of  $y$  in  $x$  is replaced by  $z$ . If  $y \in X_0 \cap \wp(x)$  and  $\neg R(x, z)$ ,  $x^{y/z}$  will be called an *irrelevant*

---

47. There are two ways of doing this: by adding a infinite sequence of symbols into  $X_0$  or a sufficient but finite supply of symbols to  $L$  without the production rules applying to them.

*substitution*, as the formula  $z$  is irrelevant to the formula it is being substituted into. If the formula being substituted,  $z$ , is not a member of  $X_0$  and the formula being substituted for is a proper subformula of the formula it is being substituted into – I will call this a *non-trivial irrelevant substitution*. This will form an important part of the theory.

A *complex* is a formula that can not be decomposed using a non-trivial irrelevant substitution.

i.e.  $x$  is a complex if

$$\neg \exists y \in \wp(x) - \{x\} - X_0; \neg R(x^y/c, y), \text{ where } c \in X_0 - \wp(x). \quad (\text{Defn } \mathbf{Cp})$$

where the sufficiency of symbols assumption above (**Suff Symbols**) ensures that such a  $c$  can be found. The idea is that if  $x \notin \mathbf{Cp}$  then there is a proper subformula,  $y$ , (that is not a member of  $X_0$ ), so that  $x$  can be thought of as the join (by substitution) of two, mutually irrelevant parts:  $x^y/c$  and  $y$ .

Let the set of all complexes in  $\mathbf{L}$  be called **Cp**. Note that trivially all members of  $X_0$  are complexes since  $\wp(x) - \{x\}$  is empty.

Following Krantz et al. in [271] I take a *measure* on  $\mathbf{L}$  to be a homomorphism from a structure on  $\mathbf{L}$  (e.g.  $\langle \mathbf{L}, \oplus, \leq \rangle$ ) into a suitable structure on the reals (e.g.  $\langle \mathfrak{R}^+, +, \leq \rangle$ ), where  $\oplus$  is some concatenation operator defined on  $\mathbf{L}$ , and  $\leq$  is an ordering relation on  $\mathbf{L}$ . For the structures given immediately above this is a function  $\mathbf{C}: \mathbf{L} \rightarrow \mathfrak{R}^+$ , defined on  $\mathbf{L}$  into the non-negative real numbers such that, for all  $x, y \in \mathbf{L}$ :

$$\mathbf{C}(x \oplus y) = \mathbf{C}(x) + \mathbf{C}(y)$$

$$x \leq y \Leftrightarrow \mathbf{C}(x) \leq \mathbf{C}(y).$$

I will use ' $\leq$ ' for both the ordering on  $\mathbf{L}$  and the normal ordering on the reals. Which I am referring to should be clear from the content. Similarly I will write  $x < y$  where  $x \leq y$  but not  $y \leq x$  and  $x \sim y$  if  $x \leq y$  and  $y \leq x$ .

In all below assume  $x, y, z \in \mathbf{L}$  and  $c, k \in X_0$ .

### 5.3.2 Weak complexity measures

There is no natural and general concatenation operator on  $\mathbf{L}$  since, in general, how and where expressions are joined matters (due to the tree structure of members of  $\mathbf{L}$ ). Substitution is a general and natural operation, but this is essentially a ternary operation.

However, there is a special case of substitution that can act in a similar way to a concatenation operator, this is the case of irrelevant substitution. This corresponds to the intuitions discussed in section 5.2.6 on page 90 above. This concatenation would be defined:

$$x \oplus y = x^c/_y, \text{ where } c \in X_0 \cap \wp(x), \neg R(x,y)$$

but clearly  $x \oplus y$  is not uniquely defined as an operation in  $L$  since there can be more than one  $c \in X_0 \cap \wp(x)$ . However the intention is that the complexity of the result of *any* irrelevant substitution for a member of  $X_0$  will be the sum of the complexities of the original formula and the formula substituted into it. In this way irrelevant substitution performs a similar role to concatenation with respect to a *complexity* measure. For clarity and to ensure that the measures are well defined I will retain the use of explicit substitution (e.g.  $x^c/_y$ ) within  $L$  (instead of  $\oplus$ , which I will reserve for use as an implicit relation as in  $x \oplus y = z$ , by which I will mean that  $z$  is the result of an irrelevant substitution of  $y$  into  $x$ ). The idea of a measure will remain that of a homomorphism between  $+$  in  $\mathfrak{R}^+$  and any irrelevant substitution in  $L$ .

Translating the above formalisation of a measure using the substitution operation we get: a *weak complexity measure* is a homomorphism from  $\langle L, \oplus, \leq \rangle$  into  $\langle \mathfrak{R}^+, +, \leq \rangle$ , that is a function  $C: L \rightarrow \mathfrak{R}^+$ , defined on  $L$  to the non-negative real numbers such that,  $\forall x, y \in L$ :

$$c \in X_0 \cap \wp(x), \neg R(x,y) \Rightarrow C(x^c/_y) = C(x) + C(y) \quad (\text{Irrel Subs})$$

$$x \leq y \Leftrightarrow C(x) \leq C(y). \quad (\text{Subform})$$

To start with I will impose no constraints upon the ordering relation on  $L$  (other than those implied by Irrel Subs).

### Some simple results

Irrel Subs implies that the measure of all nullary symbols ( $X_0$ ) to be zero, that is:

$$c \in X_0 \Rightarrow C(c) = 0 \quad (\text{Nullary Zero})$$

Proof:

Choose  $k \in X_0, k \neq c$

$$\begin{aligned} C(k) &= C(c/k) && \text{as } c/k = k \\ &= C(c) + C(k) && \text{by Irrel Subs} \end{aligned}$$

.....

This means that a complexity measure cannot be a positive measure, i.e. it is *not* true to say that:

$$x \oplus y > x.$$

An alternative form of irrelevant substitution is sometimes useful:

$$y \in \wp(x), \neg R(x^y/c, y), c \in X_0 - \wp(x) \Rightarrow C(x^y/c) = C(x) - C(y) \quad (\text{Alt Irrel Subs})$$

Proof:

$$\begin{aligned} C(x) &= C(x^y/c/y) && \text{as } x = x^y/c/y \\ &= C(x^y/c) + C(y) && \text{by Irrel Subs.} \end{aligned}$$

.....

If  $x$  and  $y$  are essentially the same formula but with different  $X_0$  symbols (I will write  $x \approx y$ ) then they have the same complexity (as argued for in section 5.2.1 on page 87 above).

Formally this is equivalent to repeated applications of:

$$c \in \wp(x) \cap X_0, k \in X_0 - \wp(x), \neg R(x, k) \Rightarrow C(x^c/k) = C(x) \quad (\text{Symbol Change})$$

Proof:

$$\begin{aligned} C(x^c/k) &= C(x) + C(k) && \text{by Irrel Subs} \\ &= C(x) && \text{by Symbol Zero.} \end{aligned}$$

.....

**Decomposition (Decomp)**

The first result, suggested by Irrel Subs, is that any formula can be decomposed into complexes. i.e. for any  $x \in L$  there are  $a_0, \dots, a_n \in Cp$  and  $c_1, \dots, c_n \in X_0$  such that

$$x = a_0^{c_1/a_1 \dots c_n/a_n}$$

and for  $i < n$ ,  $\neg R(a_0^{c_1/a_1 \dots c_i/a_i}, a_{i+1})$  (so the above could be written as the relation:  $x = a_0 \oplus \dots \oplus a_n$

and so

$$C(x) = C(a_0) + \dots + C(a_n).$$

Note that this decomposition is not necessarily unique.

Proof:

By simple induction on length of  $x$  using a non-trivial irrelevant substitution when  $x \notin Cp$ . See section 9.3 on page 169 for this.

.....

The length of a decomposition is defined as the number of irrelevant decompositions that are required (i.e.  $n$  in the above).

A weak complexity measure is determined by its values on the complexes,  $Cp$ . That is, a function,  $g$ , on  $Cp$  in  $L$ ,  $g: Cp \rightarrow \mathfrak{R}^+$ , such that:

$$x \in X_0 \Rightarrow g(x) = 0,$$

$$x \approx y \Rightarrow g(x) = g(y)$$

$$x \leq y \Rightarrow g(x) \leq g(y).$$

will generate a unique complexity measure,  $C: L \rightarrow \mathfrak{R}^+$ , on  $L$  via:

$$C(x) \equiv_{df} g(a_0) + g(a_1) + \dots + g(a_n),$$

for some decomposition:  $x = a_0^{c_1/a_1 \dots c_n/a_n}$ , into complexes:  $a_0, \dots, a_n \in Cp - X_0$ ,  $n \geq 0$ , as above (unless  $x \in X_0$ , in which case it is its own decomposition) and hence extend the ordering,  $\leq$ , from  $Cp$  to the whole of  $L$ , via:

$$x \leq y \Leftrightarrow C(x) \leq C(y).$$

Proof:

Proof is by induction on the maximum size of formulas. In the induction step there are two things to prove: that  $C$  is well defined and that the Irrel Subs condition holds. The first is shown using a lemma that one of  $\neg R(x,y)$ ,  $y \in \wp(z)$  or  $z \in \wp(y)$  holds. The second then comes out as a consequence of decomposition into complexes. Details can be found in section 9.4 on page 170.

.....

Of, course the reverse is trivially true: a weak complexity measure on  $L$  results in a generating function as above by simple restriction of its domain to  $Cp$ . Thus there is a one-to-one correspondence between weak complexity measures and such functions on the complexes of  $L$ .

I can now demonstrate the existence of (non-trivial) weak complexity measures by generating such a measure from the function  $g: Cp \rightarrow \mathfrak{R}^+$  defined:

$$x \in X_0, u \in X_1 \Rightarrow g(ux) \equiv_{df} 1,$$

$$\text{Otherwise} \Rightarrow g(x) \equiv_{df} 0.$$

This generates a complexity measure which “counts” the number of instances of unary symbols in a formula that can be decomposed out by a non-trivial irrelevant substitution. So  $C(c)=0$ ,  $C(uc)=1$ ,  $C(u(uc))=2$ ,  $C(u(u(uc)))=3$ ,  $C(u(u(u(uc))))=4$  etc. but  $C(bc(u(u(u(uc))))=0$ . Thus by simply increasing the size of the formula in this uninteresting way we can increase its complexity – as discussed above (section 3.4.1 on page 57) this is not very satisfactory,

**5.3.3 Weak complexity measures where simple repetition does not increase complexity**

I now add a condition so as to distinguish more sharply weak complexity measures from those concerned with size, by ruling out measure such as the example above. The extra conditions are:

$$c, k \in X_0, c \neq k, u \in X_1, b \in X_2$$

$$uc \leq c \quad \text{(Unary Zero)}$$

$$bck \leq c \quad \text{(Binary Zero)}$$

Thus

$$C(uc) \leq C(c) = 0 \text{ and } C(bck) \leq C(c) = 0 \text{ so } C(uc) = C(bck) = 0. \quad \text{(Zero)}$$

This has the consequence that the complexity of an irrelevant join of expressions using a binary connective is the sum of the complexities of its parts:.

$$\neg R(x, y) \Rightarrow C(bxy) = C(x) + C(y) \quad \text{(Irrel Join)}$$

Proof:

$$\text{Choose } c, k \in X_0 \text{ such that } \neg R(x, c), \neg R(x, k), c \neq k \quad \text{(Suff Symbols)}$$

Then

$$\begin{aligned} C(bxy) &= C(bx(c^c/y)) \\ &= C((bxc)^c/y) && \text{as } c \text{ can not occur in } x \\ &= C(bxc) + C(y) && \text{by Irrel Subs} \\ &= C(b(k^k/x)c) + C(y) \\ &= C((bkc)^k/x) + C(y) && c \neq k \\ &= C(bkc) + C(x) + C(y) && \text{by Irrel Subs} \\ &= C(x) + C(y) && \text{by Zero.} \end{aligned}$$

.....

Similarly

$$C(ux) = C(u(c^c/x)) = C((uc)^c/x) = C(uc) + C(x) = C(x)$$

By simple induction we can see that any formulas where there are no repetitions of the  $X_0$  symbols that occur in it then it is has zero measure. Thus the above conditions meet the requirement that expressions with no repetitions are simple that resulted from the arguments in section 5.2.3 on page 88.

As above, a weak complexity measure under the extra ‘zeroing’ conditions is determined by its values on the complexes, **Cp**.

A function, **g**, on **Cp** in **L**,  $g: Cp \rightarrow \mathfrak{R}^+$ , such that:

$$x, y \in X_0, u \in X_1, b \in X_2 \Rightarrow c(x)=0, c(ux)=0, c(bxy)=0$$

$$x \approx y \Rightarrow g(x)=g(y)$$

$$x \leq y \Rightarrow g(x) \leq g(y).$$

will generate a unique complexity measure,  $C: L \rightarrow \mathfrak{R}^+$ , on **L** via:

$$C(x) \equiv_{df} g(a_0) + g(a_1) + \dots + g(a_n),$$

for some decomposition:  $x = a_0^{c_1}/a_1 \dots^{c_n}/a_n$ , into complexes:  $a_0, \dots, a_n \in Cp-X_0$ ,  $n \geq 0$ , as above (unless  $x \in X_0$ , in which case it is its own decomposition) and hence extend the ordering,  $\leq$ , from **Cp** to the whole of **L**, via:

$$x \leq y \Leftrightarrow C(x) \leq C(y).$$

Proof:

Proof is identical to the generation theorem above, except for the extra condition on the generating function, **g**, which transfers directly to the resulting complexity measure, **C** – details of the previous proof can be found in section 9.4 on page 170.

.....

An example of this type of measure is generated from the function  $g: Cp \rightarrow \mathfrak{R}^+$  thus:

$$x \in X_0, b \in X_2 \Rightarrow g(bxx) \equiv_{df} 1.$$

$$\text{Otherwise } g(x) \equiv_{df} 0.$$

This produces a complexity measure which “counts” the number of instances of complexes of the form  $bcc$  that can be decomposed out by a non-trivial irrelevant substitution. Here  $C(c)=0$ ,  $C(bcc)=1$ ,  $C(bc(bcc))=0$ , but  $C(bk(bcc))=1$  (when  $c \neq k$ ).

### 5.3.4 Weak complexity measures that respect the subformula relation and where simple repetition does not increase complexity

There is a natural constraint on orderings of  $L$ , namely that of the subformula relation (as discussed in section 5.2.2 on page 87). So next I consider measures from  $\langle L, \oplus, \leq \rangle$  into  $\langle \mathfrak{R}^+, +, \leq \rangle$ , that is a function  $C:L \rightarrow \mathfrak{R}^+$ , defined on  $L$  to the non-negative real numbers such that,  $\forall x, y \in L$ :

$$c \in X_0 \cap \wp(x), \neg R(x, y) \Rightarrow C(x^c/y) = C(x) + C(y) \quad (\text{Irrel Subs})$$

$$x \leq y \Leftrightarrow C(x) \leq C(y). \quad (\text{Subform})$$

where  $x \in \wp(y) \Rightarrow x \leq y$  as well as  $bck, uc \leq c$ .

All the above results still hold except the generation of measures from a function on  $Cp$ , which needs to be adapted.

Such a weak complexity measure is determined by its values on the complexes, but now we do not have as much freedom to allocate values on  $Cp$ . So now a function,  $g$ , on  $Cp$  in  $L$ ,  $g:Cp \rightarrow \mathfrak{R}^+$ , such that:

$$x, y \in X_0, u \in X_1, b \in X_2 \Rightarrow c(x)=0, c(ux)=0, c(bxy)=0$$

$$x \approx y \Rightarrow g(x)=g(y)$$

$$p_0^{c_1}/p_1 \dots c_n/p_n \in \wp(q) - \{q\}$$

$$\Rightarrow \sum_i g(p_i) \leq g(q)$$

will generate a unique complexity measure,  $C:L \rightarrow \mathfrak{R}^+$ , on  $L$  via:

$$C(x) \equiv_{df} g(a_0) + g(a_1) + \dots + g(a_n),$$

for some decomposition:  $x = a_0^{c_1}/a_1 \dots c_n/a_n$ , into complexes:  $a_0, \dots, a_n \in Cp - X_0$ ,  $n \geq 0$ , as above (unless  $x \in X_0$ , in which case it is its own decomposition) and hence extend the ordering  $\leq$  to the whole  $L$ , via:

$$x \leq y \Leftrightarrow C(x) \leq C(y).$$

Proof:

Again, the proof is identical to that in section 9.4 on page 170, except that we have to check that the subformula constraint on  $\leq$  in  $L$  holds:

Now if  $y \in \wp(q) - \{q\}$ ,  $q \notin Cp$ , then by the decomposition theorem above  $y = p_0^{c_1}/p_1^{c_2}/p_2 \dots p_n^{c_n}$  for some  $p_0, p_1, \dots, p_n \in Cp$ .

$$\begin{aligned}
 \text{Now, } C(q) & \\
 &= h(q) && \text{by construction} \\
 &\geq \sum g(p_i) && \text{by (3)} \\
 &= C(y) && \text{by construction}
 \end{aligned}$$

.....

An example of this type of measure is generated from the function  $g: Cp \rightarrow \mathfrak{R}^+$  thus:

$$x, y \in X_0, u \in X_1, b \in X_2 \Rightarrow g(x) \equiv_{df} 0, g(ux) \equiv_{df} 0, g(bxy) \equiv_{df} 0.$$

$$\text{Otherwise } c(x) \equiv_{df} 1.$$

This produces a complexity measure which “counts” the number of non-trivial complexes in the decomposition of a formula. Here  $C(c)=0$ ,  $C(bck)=0$  where  $c \neq k$ ,  $C(u(ux))=0$ ,  $C(bcc)=1$ ,  $C(bc(bcc))=1$ , and  $C(b(bcc)(bcc))=2$ .

The trouble with the above method of generating a complexity measure is that it is not easy to see what generating functions would satisfy the conditions. The following alternative is a little clearer.

Any non-negative function  $h: Cp \rightarrow \mathfrak{R}^+$ , recursively generates a strong complexity measure on  $L$ , thus:

$$p, q \in X_0, u \in X_1, b \in X_2 \Rightarrow C(p)=0, C(up)=0 \text{ and } C(bpq)=0,$$

$$x \in Cp \Rightarrow C(x) = C(a_0) + \dots + C(a_n), \text{ where } x \text{ has a decomposition: } a_0^{c_1}/a_1 \dots a_n^{c_n},$$

$$\text{Otherwise } C(x) = h(x) + \max\{C(y) \mid y \in \wp(x)\}.$$

In other words  $g$  specifies the “extra” complexity assigned to a complex above the greatest complexity of any of its subformula and the complexity of decomposable formulas is built up in the normal way as the sum of the complexities of the complex in a decomposition.

Thus since any such complexity measure generates a non-negative function and a non-negative function generates such a complexity measure in reverse, there is a one-one correspondence between non-negative functions on the Complexes and these complexity measures.

### 5.3.5 Strong complexity measures

#### Relevant Join

In the above measures what I call the ‘relevant join’ property does not necessarily hold. This property is:

$$R(x,y) \Rightarrow bxy > C(x), bxy > C(y)$$

These deficiencies will be corrected with the setting up of “strong complexity” measures below. Firstly I prove that the following three conditions are equivalent on a weak complexity measure, in particular the version in (section 5.3.3 on page 100), for  $b \in X_2, x, y \in L$ :

- (i)  $R(x,y) \Rightarrow C(bxy) > C(x), C(bxy) > C(y)$  (Rel Join)
- (ii)  $y \in \wp(x) - \{x\}, c \in X_0 - \wp(x), R(x^v/c, y) \Rightarrow y < x$  (Rel Subform)
- (iii)  $y \in \wp(x) - \{x\} - X_0, x \in \mathbf{Cp} \Rightarrow y < x$  (Subform of Complex)

#### Proof:

First I show that condition (i) implies the subformula property, then prove in turn (i) $\Rightarrow$ (ii); (ii) $\Rightarrow$ (i); (ii) $\Rightarrow$ (iii); and finally (iii) $\Rightarrow$ (ii). This proof can be found in section 9.5 on page 175.

.....

Note that (as a result of the proofs in section 9.5 on page 175) any of the three conditions above implies the subformula property, namely:

$$y \in P(x) \Rightarrow y \leq x.$$

I can now define a *strong complexity measure* as a weak complexity measure together with any of the three constraints above.

Again, in an almost identical way as above, we can generate strong complexity measures from measures on  $\mathbf{Cp}$ , the set of complexes.

A function,  $g$ , on  $\mathbf{Cp}$  in  $\mathbf{L}$ ,  $g: \mathbf{Cp} \rightarrow \mathfrak{R}^+$ , such that for all  $p, q \in \mathbf{Cp}$ :

$$(1) \quad p, q \in X_0 \Rightarrow g(p)=0, g(up)=0 \text{ and } g(bpq)=0$$

$$(2) \quad p \approx q \Rightarrow g(p)=g(q)$$

$$(3) \quad p_0^{c_1}/p_1 \dots p_n^{c_n} \in \wp(q) - \{q\} \\ \Rightarrow \sum_i g(p_i) < g(q)$$

will generate a unique complexity measure,  $C: \mathbf{L} \rightarrow \mathfrak{R}^+$ , on  $\mathbf{L}$  via:

$$C(x) \equiv_{\text{df}} g(a_0) + g(a_1) + \dots + g(a_n),$$

for some decomposition:  $x = a_0^{c_1}/a_1 \dots a_n^{c_n}$ , into complexes:  $a_0, \dots, a_n \in \mathbf{Cp} - X_0$ ,  $n \geq 0$ , as above (unless  $x \in X_0$ , in which case it is its own decomposition) and hence extend the ordering  $\leq$  to the whole  $\mathbf{L}$ , via:

$$x \leq y \Leftrightarrow C(x) \leq C(y).$$

Proof:

Again, the proof is identical to that in section 9.4 on page 170, except that we have to check that the subformula constraint on  $\leq$  in  $\mathbf{L}$  holds:

Now if  $y \in \wp(q) - \{q\}$ ,  $q \in \mathbf{Cp}$ , then by the decomposition theorem above  $y = p_0^{c_1}/p_1 \dots p_n^{c_n}$  for some  $p_0, \dots, p_n \in \mathbf{Cp}$ .

Now,  $C(q)$

$$= g(q) \quad \text{by construction}$$

$$> \sum_i g(p_i) \quad \text{by (3)}$$

$$= C(y) \quad \text{by construction}$$

.....

Obviously, any strong complexity measure has a unique restriction to  $\mathbf{Cp}$  that obeys (1), (2), and (3) above. Thus there is a one-one correspondence between such functions on  $\mathbf{Cp}$  and strong complexity measures.

This time any *strictly positive* function on the non-trivial complexes  $h: \mathbf{Cp}\text{-}\mathbf{X}_0 \rightarrow \mathfrak{R}^+$ , recursively generates a strong complexity measure on  $L$ , thus:

$$p, q \in X_0, u \in X_1, b \in X_2 \Rightarrow C(p)=0, C(up)=0 \text{ and } C(bpq)=0,$$

$$x \notin \mathbf{Cp} \Rightarrow C(x)=C(a_0) + \dots + C(a_n), \text{ where } x \text{ has a decomposition: } a_0^{c_1}/a_1 \dots^{c_n}/a_n,$$

$$\text{Otherwise } C(x)=h(x) + \max\{C(y) \mid y \in \wp(x)\}.$$

Thus since any strong complexity measure generates a strictly positive function and a strictly positive function generates a strong complexity measure in reverse, there is a one-one correspondence between positive functions on the complexes and strong complexity measures.

Perhaps the most straightforward of such functions,  $h$ , mentioned immediately above is the constant function  $h(x)=1$ . This measure captures the maximum length of a chain of stepwise decompositions by (possibly relevant) non-trivial substitutions. This is analogous to the Krohn-Rhodes measure of complexity based on the decomposition of semi-groups (section 8.23 on page 148).

Other strong complexity measures include: the cyclomatic complexity of the minimal directed acyclic graph representation of formula (see section 5.4 on page 106 immediately below), and the logarithm of the number of spanning trees of the same representation (see section 8.31 on page 153).

Thus strong complexity measures have all of the properties argued for analytic complexity above in sections 5.2.1 to 5.2.7. This still leaves a lot of choice, to allow the capturing in such measures of different types of difficulty. I now look at one such in more detail.

## 5.4 The Cyclomatic Number as a Measure of Analytic Complexity

I will define a possible way of comparing the complexities of complexes. Two ways of breaking a complex up are by substituting for a (non-atomic) repeated sub-formula and by breaking the top level connective into the separate parts. For example in seeking to “break up” the formula  $(a \rightarrow (a \rightarrow b)) \rightarrow (a \rightarrow b)$  we could substitute for  $a \rightarrow b$  to make  $(a \rightarrow x) \rightarrow x$  and  $a \rightarrow b$  or break the top implication to make  $a \rightarrow (a \rightarrow b)$  and  $a \rightarrow b$ . This is not a decomposition in the above sense as both methods lose some of the essential structure of the formula (the whole is more than the sum of the parts).

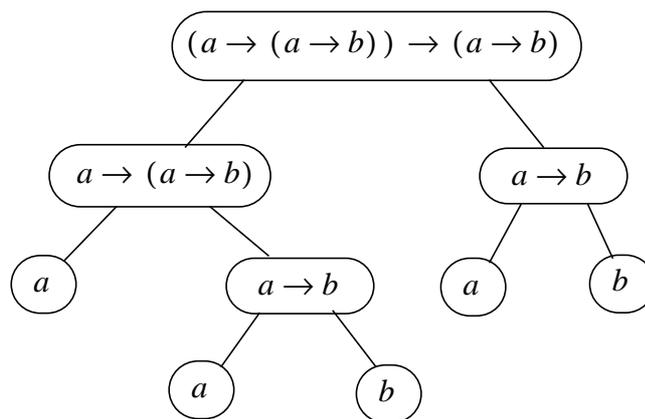
Call the damage inflicted by such a breaking the number of sub-formula the two parts now have in common. Now keep breaking the formula up until all the parts are simple. Call the total damage of such a process the “analytic loss”. Now call the complexity of the original formula the minimum such loss possible by such an “analysis”. Note that here I am seeking to talk about the analytic complexity of the formula relative to its syntax and above definition of irrelevance; I am not talking about the complexity of any particular analytic process.

For example, we could first break  $(a \rightarrow (a \rightarrow b)) \rightarrow (a \rightarrow b)$  into  $(a \rightarrow x) \rightarrow x$  and  $a \rightarrow b$  (with a damage of 1) and then  $(a \rightarrow x) \rightarrow x$  broken into  $a \rightarrow x$  and  $x$  (with a damage of 1). Thus the complexity of the formula would be no greater than 2. The only other possibilities are to first break the formula into  $(a \rightarrow (a \rightarrow b))$  and  $a \rightarrow b$  (with a damage of 3) or substitute for  $a$  (which does not progress the “analysis”. Thus the loop complexity of the formula would be 2.

This, of course is quite a coarse account of the damage inflicted in such an analysis. It says nothing of the *sort* of structural information lost in the process.

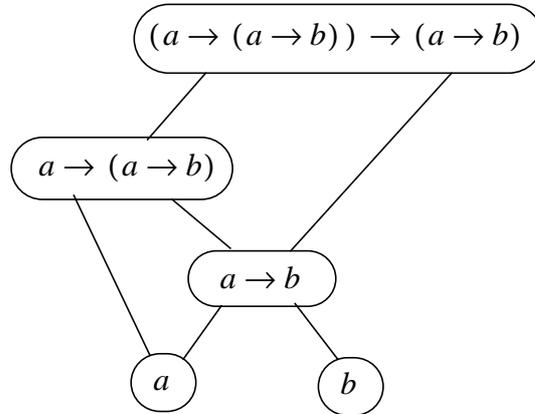
It turns out that this is bounded below by the cyclomatic number of a directed graph when the formula is represented as a minimal graph. (see section 9.2 on page 169).

For example the formula  $(a \rightarrow (a \rightarrow b)) \rightarrow (a \rightarrow b)$  can be represented by the following tree in figure 19.



**Figure 19.** A formula represented as a tree

but this is not minimal. There is a repetition of leaves that can be eliminated by combining the nodes of identical sub-formula to produce the collapsed graph shown in figure 20.



**Figure 20.** A formula represented as a collapsed tree

The cyclomatic number of this graph is then the number of connecting arcs minus the number of nodes plus one. This gives us a complexity of two. Another way of calculating this is just by counting the number of “independent” closed loops [436].

Thus this definition is consistent with all that I specified above. Simple formula can be broken with no damage and hence have a complexity of zero. If a formula can be decomposed into parts then its complexity is the sum of those parts and the complexity of a formula is at least as great as any sub-formula. See section 5.6.1 on page 113 for a comparison of complexity measures on such formulas.

One advantage of such a measure of complexity is that it is applicable to a wide range of structures and certainly any statement in a wide range of formal syntaxes.

### 5.5 Layers of Syntax and Complexity

One method frequently used (by humans) to attack such complexes is to change the language they are expressed in. A formula that may be complex in one language might be simple in another. For example, the complexity of a formula in a language’s formula syntax may have no obvious connection with the complexity of a derivation of such a formula as a proof in the proof theory’s syntax (see section 5.6 on page 113).

Thus in order to study what is still difficult within this more subtle situation we need to be able to consider varying syntaxes where it is the formula that essentially remains constant. To do this we need a formalism that will encapsulate both the formula syntaxes

as well as the syntaxes of proofs. I could do this as a generalised language defined by the production of strings of symbols but this does not clearly preserve for study the distinction between the structure of a syntactic derivation and its result. I want a very clear distinction between the structure of a derivation (like a proof) and its result (a formula) as these belong to *different* syntaxes. Also a linear string is, in many respects, an artificial representation of formal statements, requiring the use of such devices as brackets and parsing. Thus for the purposes of this study I will define a generalised description of a substantial class of syntaxes as tree structures.

I will define a general syntactic structure recursively. Presume there is a set of primitive symbols called P and also a set, V, of variables that will be used in the definition of its generative rules. A syntactic structure, S, is then a pair (X, R) where X is set of sub-syntactic structures (possibly empty) and R is a set of rules of the form:

$$\left( (A_1, \dots, A_n), C, \{ (v_1, X_1), \dots, (v_m, X_m) \} \right),$$

where the  $A_i$  are trees with the nodes labelled with primitive symbols or variables from the syntax,  $C$  is the result of the rule, the  $v_i$  are variables and the  $X_i$  are members of X - the "lower" syntaxes that this one draws on for formulas, symbols etc. In addition there is the restriction that every symbol appearing in  $C$  must either have appeared in one of the  $A_i$ , be a symbol from P or be one of the  $v_i$ .

The idea is that if the  $A_1, \dots, A_n$  occur in the syntax then  $C$  will as well, with any occurrences of  $v_i$  being replaced with an item from the sub-syntax generated by  $X_i$ .

For convenience I will write the rule like this:

$$A_1, \dots, A_n \gg C, (v_1 \in X_1, \dots, v_m \in X_m) .$$

Such a syntactic structure will generate a syntax being the set of all trees recursively constructible from the rules.

I will make this clearer with some examples. In these let the set of primitive symbols, P, include x,  $\rightarrow$ ,  $\neg$  and V include u, v, w.

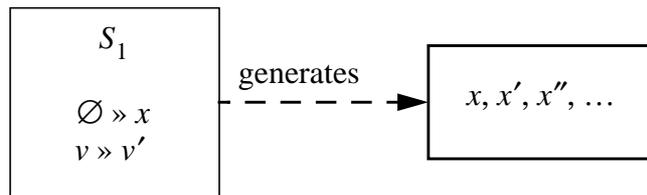
### 5.5.1 Example 1 - a supply of variable names

Let  $S_1$  be the syntax  $(\emptyset, R_1)$  where  $R_1$  consists of the rules:

$$N1. \quad \emptyset \gg x$$

$$N2. \quad v \gg '(v), \text{ (or } v \gg v' \text{ when allowing suffix notation).}$$

This generates (in a way that will be specified) the set which could be written as  $\{x, x', x'', \dots\}$  but more pedantically written as  $\{N1, N2(N1), N2(N2(N1)), \dots\}$ . This could be used as an infinite supply of pairwise distinct names. There is an important distinction here between  $\{N1, N2(N1), N2(N2(N1)), \dots\}$ , which are the structures of the possible derivations and  $\{x, x', x'', \dots\}$  which could be considered as the result of performing the derivations. This is illustrated in figure 21.



**Figure 21.** A single layer syntactic structure

In this first example they can be safely conflated as they have a one-to-one structural correspondence, but in the next examples this will not hold.

### 5.5.2 Example 2 - WFFs of the implication fragment

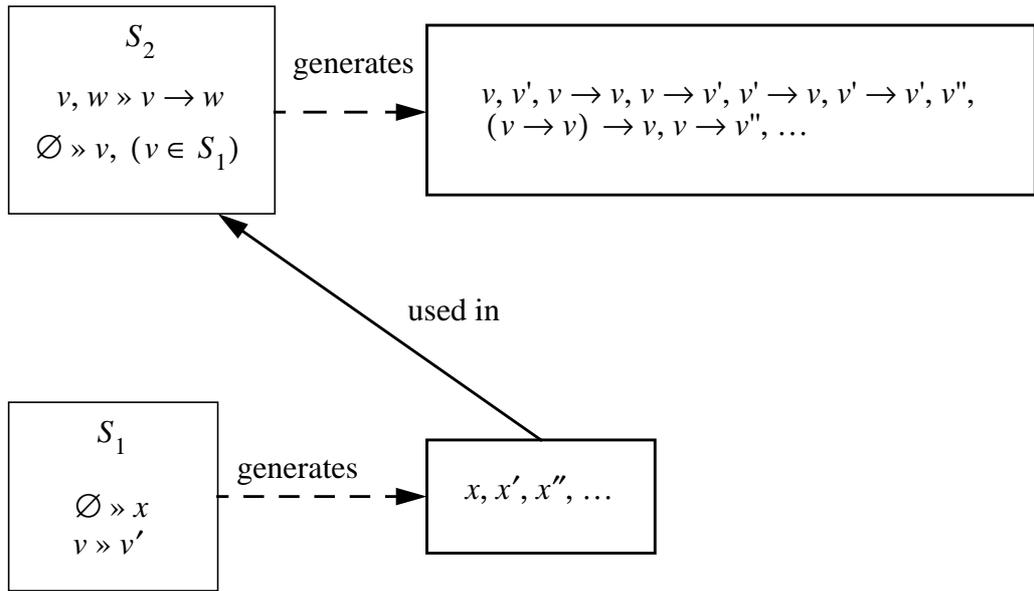
Now let  $S_2$  be the syntax  $(S_1, R_2)$ , where  $R_2$  consists of the rules

F1.  $\emptyset \gg v, (v \in S_1)$

F2.  $v, w \gg \rightarrow(v, w),$  (or  $v, w \gg v \rightarrow w$  using infix notation).

This generates the syntax of pure-implication propositional formulas, which could be written as  $\{v, v', v \rightarrow v, v \rightarrow v', v' \rightarrow v, v' \rightarrow v', v'', (v \rightarrow v) \rightarrow v, v \rightarrow v'', \dots\}$ . The first rule allows for the inclusion of trees generated by  $S_1$  and the second the recursive construction of the pure implicational formulas in the normal manner.

Here we have a two-layer system of syntactic structures, illustrated in figure 22.



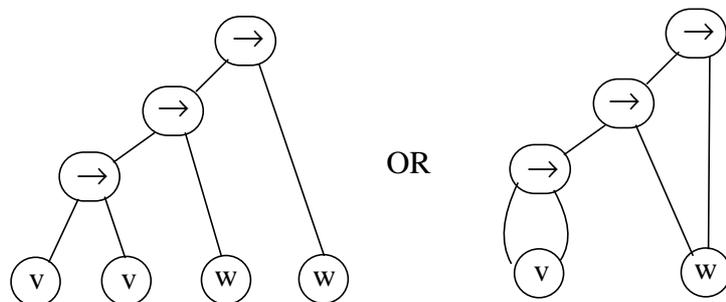
**Figure 22.** A two layer syntactic structure

**5.5.3 Example 3 - The implicational fragment of E**

Now let  $S_3$  be the syntactic structure  $(S_2, R_3)$  where  $R_3$  consists of the rules

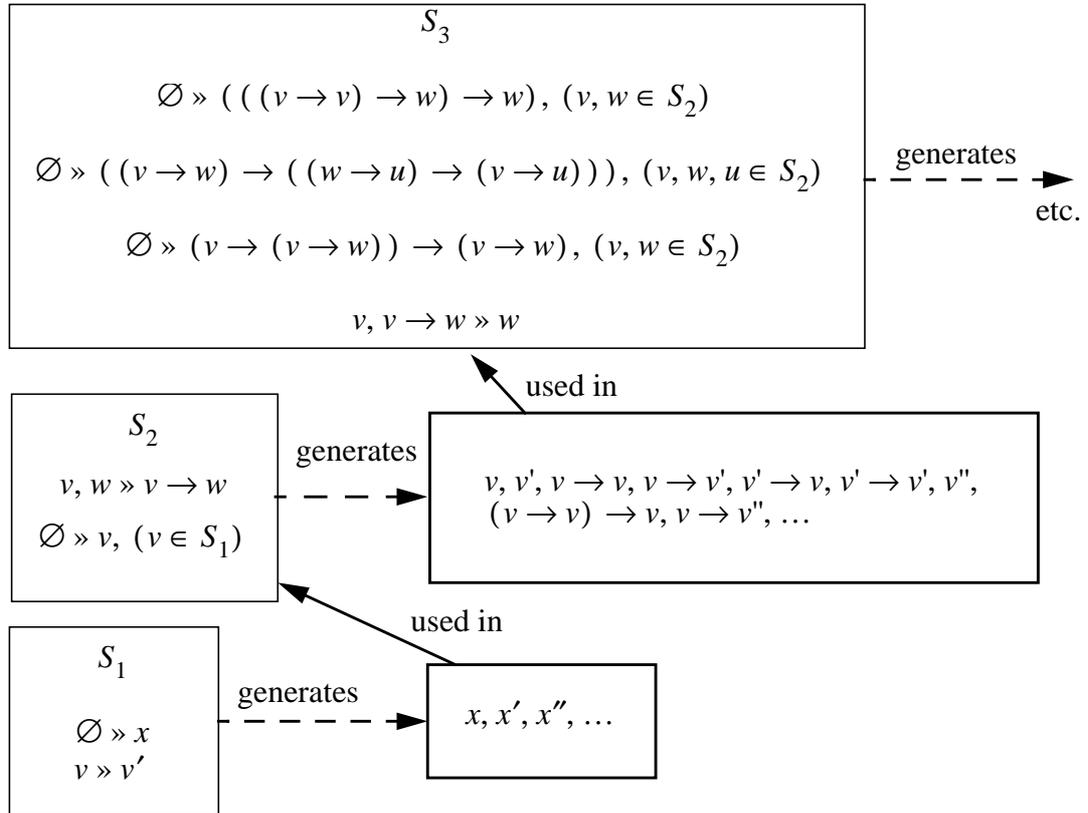
- P1.  $\emptyset \gg ((v \rightarrow v) \rightarrow w) \rightarrow w, (v, w \in S_2)$
- P2.  $\emptyset \gg ((v \rightarrow w) \rightarrow ((w \rightarrow u) \rightarrow (v \rightarrow u))), (v, w, u \in S_2)$
- P3.  $\emptyset \gg (v \rightarrow (v \rightarrow w)) \rightarrow (v \rightarrow w), (v, w \in S_2)$
- P4.  $v, v \rightarrow w \gg w$

which will generate a syntax of proofs of pure implicational tautologies. Note that the brackets in rules P1, P2 and P3 are for human convenience only.  $((v \rightarrow v) \rightarrow w) \rightarrow w$  really stands for the structure shown in figure 23.



**Figure 23.** The actual structure of the assertion axiom

In this example we have three levels of syntactic structure for constructing distinct propositional names, formulas and proofs. This is shown in figure 24.



**Figure 24.** A three layer structure

#### 5.5.4 Discussion of syntactic structures

A syntactic structure generates a syntax, which is the closure under the rules of the empty set. Thus in order to be non-empty at least one rule must have an empty set of antecedents (like the rules N1, P1, F1 or F2 above). I call such rules *leaf nodes*.

There are two kinds of common leaf nodes: those where the consequent is constructed purely from the set of primitive symbols (like rule N1 above) and those whose antecedent is constructed with a substitution from a sub-syntax (like P1, F1 or F2).

Thus the syntax is composed of a set of derivation structures constructed from the rules and specific substitutions of items from the syntaxes of sub-syntactic structures. For example the syntax generated by  $S_3$  would include the proof for  $((x \rightarrow x) \rightarrow x) \rightarrow x$  which could be represented as  $P1(v \leftarrow F1(v \leftarrow N1), w \leftarrow F1(v \leftarrow N1))$ , i.e. the rule

$P1$  with  $x$  substituted for both  $v$  and  $w$  from the syntax generated from  $S_2$ . This  $x$  is the rule  $F1$  with the rule  $N1$  substituted from the syntax generated from  $S_1$ .

A higher level derivation structure can be 'executed' to produce a result from a lower syntax. Thus the above structure  $P1(v \leftarrow F1(v \leftarrow N1), w \leftarrow F1(v \leftarrow N1))$  can be executed to the structure  $F2(F2((F2(F1(v \leftarrow N1), F1(v \leftarrow N1)), F1(v \leftarrow N1)), F1(v \leftarrow N1)))$  which I would normally write as the string  $((x \rightarrow x) \rightarrow x) \rightarrow x$ .

In section 10.1 on page 182 in Appendix 3 - Formalisation of Syntactic Structure, I formalise the above notions of syntactic structures and the generation from them. I also show (section 10.2 on page 184) that they have at least the expressive power of a general phrase structured grammar.

In Appendix 4 - A tool for exploring syntactic structures, complexity and simplification, I describe a software tool I developed for exploring such structures. It can deal with a sequence of syntactic structures, where each SS is dependent on the one below. In section 10.3 on page 187 of that appendix I outline a proof that the restriction to such a linear chain does not reduce the overall expressive power of such systems.

## 5.6 Application to Axioms and Proof Systems

### 5.6.1 Axiom complexity

Elegance has long been a desirable characteristic in formal mathematics and logic. During the first half of this century this has frequently been associated with parsimony. In particular there was a drive to find formal logical systems with the fewest rules, axioms or variables.

In Hilbert-Frege style axiomatisations [162, 222], the number of rules is reduced to two, substitution (subs) and Modus Ponens (MP)<sup>48</sup>, so that the properties of the logic are determined solely by the axioms which thus have a dual role as starting points for proofs and inference tickets (i.e. as the major premiss of an application of an MP rule). This focused attention on the number and length of the axioms.

Many different axiomatisations have been suggested. In 1910, Whitehead and Russell produced an axiomatisation of the disjunction-negation fragment using 5

---

48.Or just MP if one considers axioms as axiom schemas.

axioms [467] (later one was shown to be a consequence of the others)<sup>49</sup>. All the other connectives were introduced as definitions of these.

This was followed by many other suggestions, many of which favoured parsimony. In 1929 Lukasiewicz produced an axiomatisation of the implication-negation fragment of classical logic using just three axioms<sup>50</sup>:

- (1)  $(p \rightarrow q) \rightarrow ((q \rightarrow r) \rightarrow (p \rightarrow r))$  ;
- (2)  $(\neg p \rightarrow p) \rightarrow p$  ;
- (3)  $p \rightarrow (\neg p \rightarrow q)$  .

In 1949 he exhibited a single axiom system for the pure implicational fragment, proving its minimality [300]:

- (4)  $((p \rightarrow q) \rightarrow r) \rightarrow ((r \rightarrow p) \rightarrow (s \rightarrow p))$  ,

followed by several systems by Meredith [320, 321] in the 50's including a single axiom version of the implication-negation fragment:

- (5)  $((((p \rightarrow q) \rightarrow (\neg r \rightarrow \neg s)) \rightarrow r) \rightarrow t) \rightarrow ((t \rightarrow p) \rightarrow (s \rightarrow p))$  .

There was also an investigation into the minimal number of connectives used. In 1918 Nicod exhibited a single axiom with a single connective equivalent to  $\neg(p \wedge q)$  involving 11 occurrences of that connective and 5 variables<sup>51</sup>.

Similarly there was also a drive to reduce the number of variables used. Lukasiewicz "improved" upon Nicod's axiom by finding one of the same length but only involving 4 variables<sup>52</sup>. Later Diamond and McKinsey showed that one needed at least one axiom with three variables in it [139] and recently Ulrich proved more about the necessary length of axioms using matrix techniques [448].

Perusing the above it is obvious that mere parsimony does not correspond to any relevant sense of complexity, with respect to such axiomatisations. While Lukasiewicz's *three* axioms above are amenable to a natural interpretation (versions of transitivity, necessity out and inferring from a contradiction), this is very difficult to do for the single axiom systems. Thus going by the definition of complexity herein the only way in which they could be said to be simpler is due to the difficulty of storage of the axiomatic

---

49.Frege had already exhibited this type of system [162], but his notation was cumbersome.

50.Listed using the standard infix style notation rather than Lukasiewicz's.

51.I will not show this, suffice to say it is incomprehensible!

52.As reported by Prior in [362].

description given the logical theory it needs to produce, and the difficulty of storage must be one of the least of one's problems with such systems.

In appendix 4 (section 12 on page 193) I have tabulated all the propositional formulas in the implication-negation fragment up to those with 6 symbols (not including brackets), sorted by five methods: size, number of variables, depth, breadth and cyclomatic number. All these rankings give credibly low rankings to simple formulas but can differ widely on the rest.

Table 2 on page 193 shows these formulas sorted by size. Here formulas such as  $\neg\neg\neg\neg\neg a$  or  $a \rightarrow (b \rightarrow c)$  are ranked highly, despite their analytical straightforwardness. In general, as with other size based measures of complexity, size can be important if there are sharp limitations on storage (such as short term memory), but otherwise do not usually pose the most difficult problems.

In Table 3 on page 194, we see the ranking by the number of (distinct) variables. This will always rank formulas of the form  $a \rightarrow (b \rightarrow (c \rightarrow \dots z))$  the furthest down the list. In particular here,  $(\neg a \rightarrow a) \rightarrow a$  is lower than  $a \rightarrow (b \rightarrow c)$ . The number of distinct variables has been closely linked to measures of *simplicity*. Kemeny [254] and Goodman [186] elaborate this by also introducing extra criteria concerning symmetry. I argue that complexity is sensibly distinguished from such measures which are perhaps better thought of as rough measures on information (see section 6.5 on page 129).

Table 4 on page 195 looks at the formulas in terms of their (maximum) depth. Clearly depth (as to a lesser extent size) can impose difficulties in terms of computation both in inference and induction. Nevertheless formulas like  $\neg\neg\neg\neg\neg a$  appear well down the list. Such a ranking may be important if the operator (in this case negation) itself implies a more complex interaction (such as a modal operator might imply between possible worlds).

The breadth of formulas is shown in Table 5 on page 196. This is a weak measure of complexity as the formulas can be indefinitely large, deep and still be fairly interconnected. Primarily this could be seen as a limit on the input to the formulas, or alternatively representing a limit on a hypothetical top-down search on the structure of the formulas. A formula like  $((a \rightarrow a) \rightarrow (a \rightarrow a)) \rightarrow ((a \rightarrow a) \rightarrow (a \rightarrow a))$ , which intuitively represents merely a multiple use of substitution in identity will have maximal

breadth, but it is analytically simple (this would not help you, of course, if you have to process it top-down in ignorance of the content of its leaves).

Finally Table 6 on page 197 shows them ranked by the cyclomatic number of the formula's minimal directed acyclic graph (as explained in section 5.4 on page 106). Again formulas can be of indefinite size and depth, for a given ranking, but only if a suitable number of distinct variables are introduced (or if you are merely iterating negation). Formulas of the form  $a \rightarrow (b \rightarrow (c \rightarrow \dots z))$  are always maximally simple in this way, representing the fact that the formula *in itself* does not encode any complexity of relationship. Non repetitive but involved formulas referring to variables repeatedly but in different ways come out as complex.

The different rankings of a selection of formulas are then directly compared in Table 7 on page 198. The cyclomatic number of the graph of the references in a formula gives a much better measure of an intuitive idea of their *analytic* complexity than the others.

Of course, as I have repeatedly stressed above, the complexity of an expression depends on its context (composed of language, goals and viewpoint). Thus it is important, for example, to keep in mind whether one is talking about the complexity of interaction encoded as facts about the natural world represented in formulas or talking about the properties of the connectives themselves by using axioms and the like. The difference comes out particularly in the appropriate relevance relation: if you are talking about the natural world then the relevance between identically named variables in separate formulas reflects the uniform reference intended by these formulas; if you are implicitly specifying the properties of the connectives (as in formal logic) then the appropriate reference is also intended between identical connectives between the axioms. Thus in these two different situations a different relevant relation needs to be used and this would affect the model of analytic complexity.

### 5.6.2 Proof complexity

Systems with minimal axioms are very difficult to use to prove anything. The few but flexible rules mean that the form of the desired theorem gives one little indication as to how to proceed. The proof, of identity, using (4) is shown in figure 25<sup>53</sup>.

1	CCCpqrCCrpCsp	<i>Axiom</i>
2	CCCCpqrCCrpCspCCCCrpCspCpqCrCpq	<i>Subs. inst. of 1</i>
3	CCCCrpCspCpqCrCpq	<i>MP 1, 2</i>
4	CCCCrpCspCpqCrCpqCCCrCpqCCrpCspCtCCrpCsp	<i>Subs. inst. of 1</i>
5	CCCrCpqCCrpCspCtCCrpCsp	<i>MP 3, 4</i>
6	CCCPqCpqCCcpqCsp	<i>Subs. inst. of 1</i>
7	CCCCpqCpqCCcpqCspCCCCpqrCCrpCspCCcpqCsp	<i>Subs. inst. of 5</i>
8	CCCCpqrCCrpCspCCcpqCsp	<i>MP 6, 7</i>
9	CCcpqCsp	<i>MP 1, 8</i>
10	CCCPqpCspCCspCpqCrCpq	<i>Subs. inst. of 1</i>
11	CCspCpqCrCpq	<i>MP 9, 10</i>
12	CCcpqCsp	<i>Subs. inst. of 9</i>
13	CCCPqpCspCCcpqCspCsp	<i>Subs. inst. of 11</i>
14	CCcpqCspCsp	<i>MP 12, 13</i>
15	Csp	<i>MP 9, 14</i>

**Figure 25.** The proof of identity in Lukasiewicz's single axiom system

In practice logicians tend to deal with the difficulty of producing proofs in these circumstances by developing an elaborate system of derived rules. Thus they implicitly change both their language of representation of the problem of producing such proofs in terms of these derived rules as well as the original axioms and rules. The greater simplicity of the problem is a result of an implicit change in the language of representation.

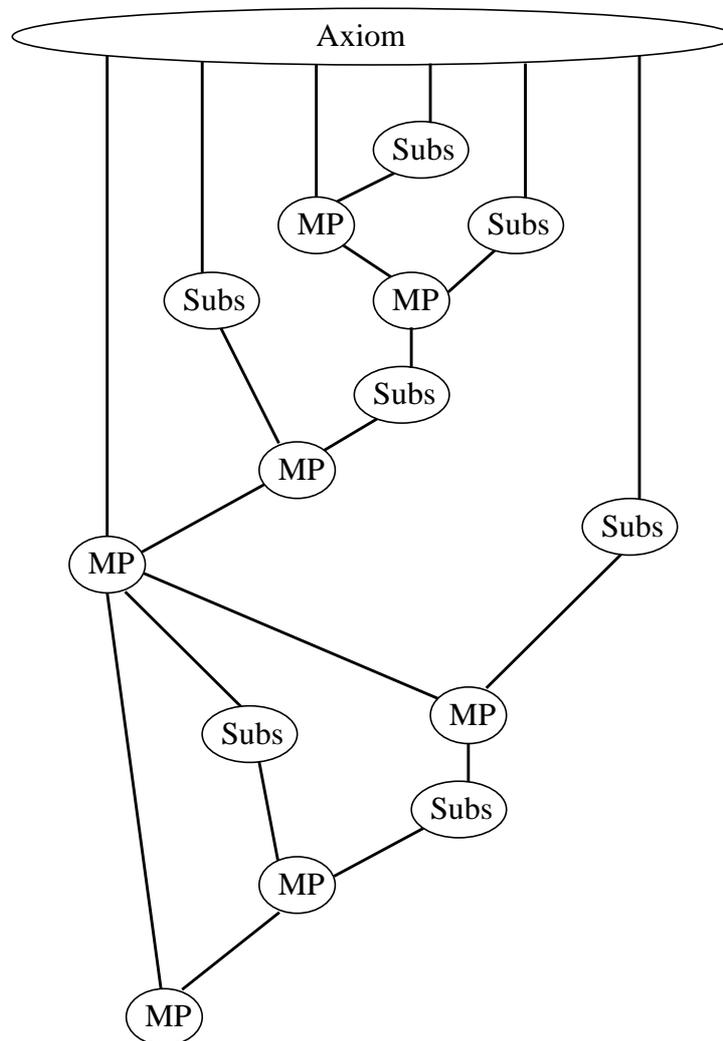
For any particular theorem, however complicated, one could trivially invent a proof system where its proof was simple by adding it as an axiom. Likewise for any theorem,

---

53. The proof is displayed using polish notation and using a substitution rule instead of axiom schemas in order to shorten it.

however simple, one could add intermediate steps necessary to make its proof as difficult as wanted. The complexity of proof production is thus a separate matter from the analytic complexity of individual formulas.

Thus, in this section I concentrate on characterisations of the complexity of the proof tree of a theorem. This is an abstraction of the uses of the axioms and inference rules, where the axioms are the leaves and the inference rules the nodes. For example the proof immediately above (figure 25 on page 117) could be represented as in figure 26 below.



**Figure 26.** An example proof net

The complexity of the proof would alter dramatically if we included and related the formula that were introduced via the substitution rules. Here we have a case where the complexity of the derivational structures in the proof syntax would be more complex if we

also included the derivations of the formulas used in the substitution steps. Of course, for some purposes this would not be relevant to the task at hand, but it would be for others where one was trying to trace the referents as they were manipulated by the proof rules.

In many proof systems attempts at proof construction are made incrementally, in either a top-down fashion, starting at the target theorem or bottom-up fashion, starting with the axioms (or, of course, a combination of these). In this case the search space of potential proof trees grows exponentially with the depth. Thus there is a strong connection between the depth of the proof tree and the computational complexity of a “brute force” automatic search for a such a proof.

An analytic approach to constructing such a tree may, of course, proceed in a more “intelligent” fashion, by analysing the possible proof choices. Thus the analytic complexity of such a proof will, as in the consideration of analytic complexity of axioms etc., depend upon the ability to decompose a proof into separate subtrees. The tractability of such a tree will depend upon the relevance relation between the nodes of such a tree. Different kinds of analysis will lead, as ever, to different measures of its analytic complexity. For example if the relevance relation was expressed just in terms of which axioms were used in the sub-proof tree then one would get a different measure than if one also included what formulas were used to be substituted into those axioms (which would be different again from the situation where every different instance of an axiom schema was considered as different).

Different kinds of proof procedure exhibit different degrees of decomposability in the proof production process. A Hilbert system with the minimal number of axioms will thus produce proofs that are more analytically complex than, for example, those produced using a Fitch based procedure, where the proof of intermediate steps can be recursively constructed to *some extent* independently to the rest of the proof. Of course, such a decomposition of the proof process may, in some circumstances, be illusory in that the sub-task may be as hard (or even harder) to prove than the original. This does not prevent Fitch-style procedures aiding considerably in the decomposition of proof tasks for a wide range of systems and for a wide range of target theorems. Thus the claim that Fitch proofs are simpler is borne out, even if in some cases (e.g. in  $\mathcal{R}$  due to the undecidability of theoremhood there) it may not help with the most difficult cases.

Even if you restrict yourself to Hilbert style systems the set of axioms chosen can make a difference to the complexity of proof. For example, Robinson [380] exhibits a “nice” set of axioms composed of groups of axioms for each connective, such that you only have to use the axioms that correspond to the connectives in the target theorem. This is in contrast to many systems, for example in many classical axiomatic systems one has to use the properties of negation to prove the theorem  $((a \rightarrow b) \rightarrow a) \rightarrow a$ , even though it does not mention negation. In this way Robinson’s axioms allow the partial partitioning of the proof space by the type of connective. This partitioning represents an analytic simplification of the proof process.

## 5.7 Application to Simplification

In this section I briefly look at five possibilities for systematic simplification given the above analyses of complexity: those of searching for equivalent expressions; searching for equivalent derivations; by specialising the syntactic level in the underlying language; by more generally searching over different languages; and by sacrificing accuracy or specificity.

### 5.7.1 Searching over equivalent expressions within a language

The most obvious way to proceed in simplification is to attempt to search all acceptable expressions in some language for the most simple one. So one might look through all proofs starting with the simplest and working up (in terms of depth) until one arrived at one which proved the desired theorem, which would then be the simplest. Of course a slightly more intelligent procedure would involve a lot of intelligent pruning of the search space before hand, but unless this pruning can be arranged so that it drastically reduces the search space, such a procedure would have limited practical applicability.

In some cases this is not only impractical but impossible. The question is given some formal language,  $L$ , some equivalence relation,  $\sim$ , defined on  $L$ , some measure  $C:L \rightarrow \mathfrak{R}^+$ , and a specific expression,  $l \in L$ , is there a program that can always find a expression  $m \in L$ , such that  $m \sim l$  with  $C(m)$  minimal? In general the answer is, unsurprisingly, “no”. For example, it is well known that the problem of finding the minimum sized program for computing a pattern is uncomputable [286] (thus in this case  $L$  is the space of programs,  $C$  is their length and  $m \sim l$  iff  $m$  and  $l$  compute the same pattern).

Even if such a question was computable for some  $L, C$  and  $\sim$  there will typically be no way of avoiding searching the space of available expressions  $L$  and checking them all (unless  $P=NP$ , which is considered unlikely [231]).

### 5.7.2 Searching over equivalent derivations within a language

Part of the problem with the above approach is the possible ineffectiveness of the equivalence relation,  $\sim$ . If we have step-wise procedures for computing equivalent expressions the task becomes easier. For example, if one were seeking to simplify an algebraic expression like  $2x^2(yz - y^3) + (xy - 3)(1 - xyz)$  one would have a range of possible transformations one could apply to the expression based on the properties of simple arithmetic: factorisation, expansion, association, permutation of commutative operators etc.

This would make the search procedure considerably more tractable and several effective techniques would become available. One such is genetic programming [270]. This is a version of the standard genetic algorithm [228], but where the genes that are evolved according to an evaluation of fitness are tree-structured expressions instead of merely strings of a fixed length. In the case of arithmetic simplification the trees that would be evolved would be possible applications of transformations to the original expression with the fitness of such a gene being decided by the simplicity of the result. The algorithm proceeds by starting with a random population of such compound transformations represented as tree-like structures, then evaluating these expression's fitness (some inverse of complexity) and creating the next population by a combination of propagation and a sexual operator called "crossover"<sup>54</sup> on parents chosen probabilistically in proportion to their fitness. Genetic programming techniques have been used in many domains and there are now many refinements and variations on the technique (see for example [15] and [261]).

### 5.7.3 Specialising the syntactic level

Consider the following 1-level syntactic structure for generating the propositional implication-negation wffs:  $S=(\emptyset, \{R1,R2,R3,R4\})$  where rules  $R1, \dots, R4$  are defined as follows.

---

54. In this a random subtree is chosen inside each parent and then these are swapped around.

R1:  $\emptyset \gg v$

R2:  $x \gg x'$

R3:  $x \gg \neg x$

R4:  $x, y \gg x \rightarrow y$

The first two rules produce an indefinite supply of unique variable names  $v, v', v'', \dots$  and the second two are the normal syntactic production rules for negation and infix implication.

Now if one's definition of relevance is that two expressions are relevant if they share a symbol in the syntax, then everything will be relevant to everything else because all expressions generated by  $S$  will share the symbol  $v$ . Clearly, although there is a sense in which all variables *are* related in this system, it might be *intended* that different variables (ones with a different number of primes) not be related. In this case a structure that is closer to what was intended would be  $S_2=(S_1,\{V,R_3,R_4\})$ , where  $V$  was defined as follows

$V: \emptyset \gg x (x \in S_1)$

and  $S_1 = (\emptyset, \{R_1,R_2\})$ , where  $R_1, \dots, R_4$  are defined as above. Now the generation of variable names is done by  $S_1$  and the building of formulas is done by  $S_2$ . Now by the definition of relevance different variables will not be relevant to each other in the sense above. Thus the formula  $v \rightarrow v'$  would have a cyclomatic complexity of 1 in  $S$  but of zero in  $S_2$ , i.e. changing the language from  $S$  to  $S_2$  has resulted in a simplification. Of course, this is equivalent to suitably changing the relevance relation in  $S$ , but here the separation of the syntax allows for a more natural and clearer analysis of the intended situation.

The above is a very artificial example, but such shifts of syntactic level do occur. The sentence "Throughout June I sought her in vain." will have a different complexity if you are considering the mapping between words and referents than if you look at the correspondence between spelling and phonemes (e.g. the intricacies of the "ou" spelling in English).

In logic, the formula  $(a \rightarrow b) \rightarrow (a \rightarrow b)$  is often treated as a special substitution instance of identity, for the purposes of inference, rather than as signifying something in its own right in detail.

Another example comes from chaos theory. It is well known that the ‘tent map’<sup>55</sup> can generate sequences that are indistinguishable from purely random ones. If one models this process by considering all different numbers as irrelevant to each other (i.e just another number), then this seems very surprising - we have what seems to be a simple deterministic process producing random behaviour. Closer inspection reveals that the source of randomness comes from the initialisation of the system, as almost every<sup>56</sup> (abstract) real number has a ‘random’ decimal expansion, and the tent map merely “unpacks” this expansion and returns it digit by digit.

Two other examples from the literature<sup>57</sup> are: seeking to simplify the execution of programming languages by reduction over programming language hierarchies [179] and simplifying the visual presentation of graphs by reduction and abstraction [258].

One final example is arguably this thesis itself; the pushing of much of the detail into appendices is intended to simplify the presentation of the main argument.

#### 5.7.4 Searching over equivalent languages

Perhaps the most radical but also the most natural mechanism for simplification is to change the language of representation. Changing the syntactic level (as described above in section 5.7.2 on page 121) is a particular example of this. Another example is when we enrich our proof language to simplify proofs (as described in section 5.6.2 on page 117).

Part of the problem with formalising complexity is that we humans seem to be very good at “flipping” between sub-languages in order to make things simpler. Since complexity, as I have argued, is relative to the language of representation, the complexity will change with each “flip”. Things can seem very different on the page, because we often fix the formal language (at least for the purposes of display of formal properties) and investigate the properties in that language.

In order to formalise such a procedure for simplification one needs some mechanism for fixing the reference and structure of a formula whilst changing the language it is embedded in. This can be done using the syntactic structures described above (section 5.5 on page 108). The idea is to fix the “lower” structures and vary the top one. For example if

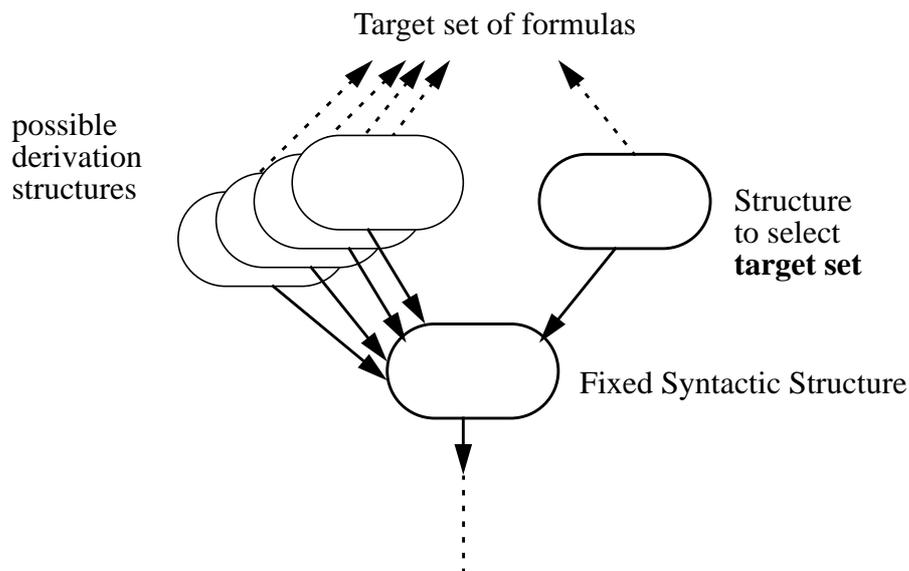
---

55.The iterated map  $x \rightarrow 2x$  if  $0 \leq x \leq 0.5$  and  $x \rightarrow 2(1-x)$  if  $0.5 < x \leq 1$  see [39] for details.

56. ‘almost every’ in the sense that it is true for every number except the rationals

57.Although these are not discussed therein from the same perspective.

one were searching for a proof system that gave as simple derivations as possible (for a target set of theorems) one could define a syntactic structure to generate all the possible expressions and another syntactic structure to select the target set. Then consider a range of equivalent syntactic proof structures that will derive the same target set. The idea is represented below in figure 27 on page 124. The solid lines represent the sub-syntax relation and the dotted lines represent generation.



**Figure 27.** A setup for varying languages

Of course, in general searching over equivalent languages will be at least as hard as searching over equivalent expressions. One could quite well represent the syntactic structures in a tree-like structure upon which a genetic programming algorithm would work, but it would frequently be computationally onerous to check that such a structure generated the same target set (and sometimes this would be totally uncomputable).

If, however, we knew of some acceptable transformations of the syntactic structure which preserved the target set we could search over the possible transformation sequences in a way similar to that outlined in section 5.7.2 on page 121. This, however is a matter well beyond the scope of this thesis.

### **5.7.5 Simplification via trade-offs with specificity and accuracy**

All the above methods of simplification maintain the same model content, which is appropriate in formal systems. However perhaps the most common methods of

simplification is where the content of the model is allowed to change so as to trade-off the simplicity of a model with the model's accuracy or specificity.

Consider first complexity-specificity trade-offs. In some cases, especially when the data can be separated into what might be seen as random noise and an 'underlying' signal, a slightly less specific model of the data can be a lot simpler. So, for example the diagram on the right in figure 14 on page 62 representing complete randomness would take a model of high complexity if it were to be modelled in every detail. If on the other hand a model of it as just a random pattern retaining only the granularity, density, and the probabilities of occurrences of black and white dots were used then this might be a lot simpler. Associating the complexity of the pattern with the complexity of its most appropriate model restores our intuitions about them. It is the insistence of an ultimately specific model of these patterns (for example via its reproduction via a Turing Machine) that leads to the unintuitive result that a random pattern holds the most information.

A model can be made less specific in several ways, the simplest being the relaxation of its predictions from a precise value to an range (as is implicit in the notation  $12.6\text{cm}\pm 1\text{mm}$ ); another way is to narrow the conditions of application of the model<sup>58</sup>. Where the distribution of the randomness can be more precisely specified this might be characterised in terms of a probabilities.

Another trade-off is the complexity-accuracy trade-off. This can have a similar effect to the complexity-specificity trade-off: here an increase in error may be acceptable if it results in a sharply simpler model, especially if it is thought that the more elaborate model might not be robust. After all, *some* level of error is deemed acceptable in almost all experimental science – we accept that it is usually not sensible to reject a theory due to the presence of a residual level of error. A famous case of this is the Michelson-Morley experiment, which despite the fact that it seemed to show a positive ether drift of about 5m/s (which was eventually explained in the 1960's), did not justify the rejection of the special theory of relativity.

An example of an explicit trade-off of accuracy for simplicity in the induction of decision trees is investigated by Bohanec and Bratko in [69].

---

58.This makes the model less specific overall because outside these conditions of application the model does not restrict what might happen, i.e. it says anything can happen then.

## 6 Philosophical Applications

### 6.1 Complexity and Relevance

Since the presence of irrelevance allows the top-down decomposition of complex models into sub-models without any loss of descriptive power, the pattern of relevance relations (or rather their absence) amongst the parts of a model limit the complexity of their interactions. Thus a first-cut at understanding the emergence of complex behaviour in models can be approached via relevance (in its widest sense). Of course, to gain a complete understanding one has to go beyond *whether* something is relevant, or even *to what extent* it is relevant, on to *how* it is relevant.

As argued by Hitchcock [224] relevance is properly considered a ternary relation relative to a framework or situation (much as I have relativised complexity). Thus in syntactic models of relevance which level of syntax the relevance relation is relative to is important. For example, there are several different kinds of relevance in logics: R and its related logics are concerned with relevance in the proof theory [12], while the relevance of Woods [479] is a relevance between formula, and the relevance of content (as described by Epstein [154]) is a relevance relation between the objects in the logic's semantics.

Changing the relevance relation will affect the language of representation of the model and hence its complexity. For example the analytic complexity of the proof in some formal logic might be different depending on whether one is considering the relevance relation to hold between repeated uses of rules and axioms or between formulas that refer to the same ground term. The relevance is inherent to the type of difficulty one is concerned with.

Reversing this, complexity could be used to define relevance, as happens when mutual information (section 8.25 on page 151) is defined in terms of algorithmic information (section 8.2 on page 136).

### 6.2 Complexity and Emergence

The definition given above (section 4 on page 72) implies that a complex system will be difficult to model in a top-down fashion. Thus for many very complex system we will not be able to find complete top-down models. In particular any models that are derived from the bottom-up description of the system will be unlikely to capture all the

top-level behaviour. So models that are used to try and capture some of a complex system's behaviour will be necessarily imperfect. That is to say that there is the potential for behaviour that one would not be able to predict given the models available. This behaviour is thus *emergent* in a meaningful way - it is not reducible to the bottom-up description of the system given the models available, as postulated by Cottrell in [116].

Conversely if we have a system that exhibits such emergent behaviour, that would mean that our model(s) of it did not completely capture such behaviour<sup>59</sup>. This may be because there is some fundamental reason why we are unable to produce such models, or merely that we happen not to have found such a model. In the first case our definition identifies the system as complex relative to the framework from which the fundamental reason comes; the more fundamental the reason the wider the framework. In the second case the system is complex relative to our current modelling techniques.

Only rarely are we in a position where we know that we have a fundamentally complex system rather than one which is complex relative to our current knowledge, because it is difficult to show conclusively that one has the best possible model. One generally can only show such things in artificially restricted abstract classes of models. In effect modelling is complex; it is almost impossible to ascribe optimality to a model given what one knows about a class of models relative to the goal of completely modelling most systems.

Such emergence is not restricted to the behaviour of natural systems. Chaitin [102] has exhibited a polynomial with a parameter which has inherently random behaviour, in that it is essentially random whether it will have a finite or infinite number of solutions for any given parameter value (where randomness is defined as algorithmic incompressibility). That is to say one can not predict via a theory whether there will be a finite or infinite number of solutions short of doing the calculation.

This modelling gap can be extended to one of many layers of description, each of which is the top-down level to the one below, as does Heylighen [219, 220]. This sort of structure prompts hierarchical approaches to complexity (e.g. Gougen [182]).

---

59. Darley [133] defines emergence as a difference in complexity at different levels.

### 6.3 Complexity and Language

From the definition I give and the fact that one typically wants to retain one's goals when tackling a problem, it is evident that one of the most powerful ways of tackling complexity is to change the modelling language (or framework). In other words, the choice of language is often critically important to determining the complexity relative to a type of difficulty.

This is something that we, as humans, are so good at that we are frequently unaware of it. For example, logicians working on a Hilbert style axiomatic system will frequently extend their mental model of the proof theory to include many derived rules of inference in order to tackle more complex proofs - it is not that the proofs have become simpler in the original "bare" axiomatic system, but simpler in this extended language.

Intuitively there is a qualitative difference between the simplification achievable by searching for equivalent expressions within a language and searching for equivalent languages to simplify the corresponding expressions (or even equivalent languages to simplify the search procedure for the simplest corresponding expressions in that language). For example, while a Hilbert style proof system may be parsimonious as to its inference rules, proving theorems using a Fitch style proof system is much easier because it is adapted to match key features of proofs (see section 5.6.2 on page 117).

There are interesting dynamics between complexity and the choice of language. One naturally seeks a language which will simplify the modelling of any particular subject matter, but when one finds such a language (and hence deem it a good representation, as below in section 6.4 on page 128) one is sometimes surprised by unexpected new behaviour in closely connected models - a serendipity of emergence that can tell you something genuinely novel.

### 6.4 Complexity and Representation

Peter Lupton [301] has claimed that the problem of *misrepresentation* can be explicated by considerations of the complexity of representations. Clearly, whether or not one thinks that this is the critical feature for the selection of representations, simplicity is one of the criteria for *useful* representation. This depends somewhat on the purpose of such a representation; clearly a simpler representation is preferable for beings of limited mental capacity, but further, even if one is seeking a representation which is as empirically

accurate as possible and one is limited to a particular extensible language then some heuristic to limit the mere elaboration of representations in favour of a more wider search is useful (see section 6.5 on page 129 below).

## 6.5 Complexity and “Simplicity”

“Simplicity” has a long philosophical history (e.g. [4, 78, 186, 253, 254, 339]) stretching back to Occam’s famous razor. It is defined by tradition rather than formally. It could be defined as “that property which leads one to select one theory rather than another with equal empirical support”, since at various stages almost every property of theories not directly related to its evidential support has been associated with the term, including: number of parameters [141], extensional plurality [186, 254], falsifiability [358], likelihood [390, 364], stability [447], logical expressive power [343] and content [185]. The idea started as parsimony being truth indicative. This is has now come full circle - Sober [418] defined “simplicity” in terms of relative informativeness; here the extent to which the answer is informative of the truth gave a measure of its simplicity.

Quine [364] could not see any a priori reason why a simpler theory should be more likely to be correct and Bunge [79] thought that there were too many types of simplicity for such a principle to be coherent. It would seem to presume that our language of modelling was inherently attuned to the universe (or even vice versa) such that there was a tendency for the more convenient simpler expressions also happened to be more likely to be correct.

This connection between “simplicity” and a lack of complexity seems to come from a dynamic account of theory development where an old theory being continually elaborated in an effort to maintain its consistency with discovered facts is replaced by a more coherent and powerful theory. The “simplicity” seems to be in contrast with the elaboration of the old theory.

These positions can be reconciled by considering the process of searching for acceptable theories, and what happens when they are unsuccessful. If the search process tends to start with simpler theories before trying more complicated ones, and will try elaborating theories before trying more radically different alternatives, then the fact that a theory has been elaborated will indicate that it has been unsuccessful - in such

circumstances another newer (less elaborated) equally supported theory might be a more productive choice.

Thus although a lack of complexity is no *a priori* indication of its truth, in circumstances where the process producing the theories is known to be an open-ended evolutionary process starting simply then limiting depth-first searches by some measure is a useful heuristic. Pearl [349] notes that this does not have to be a measure of complexity, but that any measure that limits the space of competing theories (for example sheer size) will do. According to my approach to complexity any such a measure could be a complexity measure, depending on how the search process worked. See also the discussion in Appendix 6 - Complexity and Scientific Modelling.

## 6.6 Complexity and Evolution

It is often assumed that complexity increases with evolution. If this assumption is based on the observation of the increase in the *maximum* complexity over all species (as in [71]) then this is assured merely by the fact that life (presumably) started simply and favoured some level of variety afterwards. That is complexity has increase not due to an active tendency but merely via a passive one due to the fact that there is a lower bound to it.

Evidence that either life as-a-whole or particular evolutionary branches have an active evolutionary tendency towards complexity is more mixed. It is uncertain that the average complexity over all species has significantly increased, since as well as the appearance of larger and more sophisticated organisms, many new simple organisms have appeared (partly to exploit the niches created by these larger organisms). In the case of individual evolutionary lines the evidence is that sometimes they evolve to be more complex but that certainly sometimes it goes in the opposite direction (as documented by McShea [316, 317]).

More deeply worrying for this assumption is that it is unclear why evolution should inherently favour the direction of the more complex rather than the simpler, especially since it is plausible that the simpler is cheaper and easier to maintain (as Martinez points out in [310]). There are some proposed mechanisms for complexity increase by evolution: Kauffman suggests that there is an inherent tendency towards order in large inter-related systems by mechanisms of self-organisation [249]; Arthur suggests that the competitive

co-evolution of species will result in such an increase [19, 21], Wimsatt points out that the evolution of multiple purposes for existing internal structures will tend to make the workings of an organism more complex [468] and Dawkins argues that evolvability itself will evolve [137].

A lot of the problems with this debate stem from the assumption that there is a single obvious notion of complexity that distinguishes us from “lower” species. Both Ho in [399] and myself in [147] attempt to separate out some of these different strands.

## 6.7 Complexity and Holism

Several holists (e.g. Rosen [384, 385, 386, 387, 388, 389], Morin [330, 331] or Mikulecky [323] - see section 8.17 on page 145) have laid claim to the word “complexity” to signify systems (or aspects of systems) that are irreducible. What is commonly called complexity is renamed as mere “complication”. This is not a practical claim that many systems are in practice irreducible, but a fundamental distinction between syntactically based mechanisms and complex systems with an essentially semantic nature (e.g. living organisms). Modelling *aspects* of such complex systems is not impossible, merely inevitably partial and not unique.

Such an approach to complexity can be seen as a special case of my approach namely that *complexity is that property of systems where it is impossible to model satisfactorily in a top-down manner given almost complete information about its atomic components and their interactions*. The fact that they do not recognize any *intermediate* states of complexity seems to be more political than analytical since it impedes analysis of the transition from systems that might be labelled “simple” to those that might be called “complex” (as in the evolution of life example in section 3.2.1 on page 47).

I would prefer to make an analogy with the idea and use of “infinity”. One can not say that any demonstrable thing is infinite (as opposed to potentially unbounded), but not withstanding this, infinity remains a useful abstraction linked to the credible extrapolation of real processes (as Rotman argues in [391]). The fact that models upon which one makes actual decisions are finite (either in representation or calculation) does not prevent the attribution of infinity to things being a useful guide to action. Similarly just because there needs to be a qualitatively different step to get from finite steps to the infinite does not prevent there being meaningful comparisons of size between different finite numbers. The

same applies to the sort of ultimate complexity holists posit. The fact that such ultimate complexity is unprovable for demonstrable models does not prevent its attribution being a useful guide to action, but on the other hand just because such complexity might be qualitatively different from normal scales of complexity does not mean that one can't make meaningful comparisons between them.

## 6.8 Complexity and System Identity

One of the questions that I avoided in posing my approach to complexity is that of system identity, namely how and why one identifies a certain set of parts and interactions as part of a unitary *system*. After all, every natural system (except, by definition, the universe) can be seen as merely a component of another system, and, (if one extrapolates from the progress of atomic physics) every natural system can be viewed as having sub-systems. On the other hand it is clear that not every collection can be said to have a meaningful *identity* as a system, for example the collection of a fish's gills, the constitution of the U.S. and a Martian rock.

One part of this conundrum can be illuminated by the observation that a system often is seen to have a "tighter" set of internal relationships and processes than external ones. This, however, is not a matter of the number or strength of such relationships. A useful example is a simple but much used module of communication software. It could be used by an unlimited number of different packages and components in a computer system, so the external relationships vastly outnumbered the internal ones, but this would not stop it being considered as an identifiable system.

Rather it is the complexity of the system, viewed as a relationship between its components as compared to the complexity of the system it was a part of that might indicate what could be usefully separated out as a system worth identifying. To use the above example, if the module was used repeatedly by only one process or component (or closely integrated set of processes) then one might well say that the module is more naturally categorised as a component of the system it is a part of, whereby if it was called by many different unrelated processes or components then its identification as a system in its own right becomes more natural.

One problem with using complexity in this way to aid system identification is that there is a danger of circularity: complexity is defined in terms of properties of models of

systems and systems are identified in terms of their complexity. What prevents such circularity making the definition tautologous or even contradictory is that at each stage the complexity of a system is grounded in an observer's model of it along with the type of difficulty that is relevant to the observer's goals, and the identification of systems is not completely determined by considerations of complexity except those that are relevant to the observer's modelling and goals.

Greenberg, in a related approach, uses an axiomatisation of identity with respect to indiscernible and complexes [198].

## **6.9 Complexity and Society**

One of the significant facts that must influence the way an individual inter-acts with society is that the complexity of that society will (at least nowadays and in the developed world) be beyond that individual's capacity to deal with (as pointed out by Beer [51]) or make plans about (Chadwick [98]). Luhman postulates that we create institutions precisely to filter out some of this complexity (as summarised by Bednarz in [50]) and further that even meaning might be a mechanism to compensate for this complexity [299]. Elsewhere I argue that the complexity of modelling the society one inhabits necessitates certain strategies on the part of the society's members as well as effecting the approach an external modeller might need to take [152].

Many (such as Casti) see the increasing complexity of society as a major problem, based on an inequality between the complexity of the system to be controlled (society) and the complexity of our models of it [88]. Some specify very general programmes of remedial action in terms of design (e.g. Galbraith in [166]) or type of approach [157, 158]. Others suggest smaller and more concrete steps for the reduction of legal complexity [82, 248]. A few are more optimistic, envisioning the emergence of a new society suited to such internal complexity (e.g. [132]).

Some holists stress the difficulty of modelling any aspect of society (e.g. Lyon stresses the caution needed with applying the new techniques of the "sciences of complexity" here [302]), but others (e.g. McIntyre in [314]) argue that this is overstated.

## 7 Conclusion

I have argued that attributing complexity only to models and not natural systems and by relativising its conception to the chosen framework (composed of language of modelling, identification of parts and overall behaviour and the type of difficulty that concerns one), one arrives at an analytically useful conception of complexity. This conception not only captures much of the intuitive idea of complexity it also allows the meaningful comparison of many different formulations of complexity as well as throwing some new light on some philosophical problems.

In particular I have met the five goals set out in the introduction (section 1.3 on page 19), which were.

1. *To lay bare assumptions and relativisations involved in the usage of the term.*
2. *To allow a meaningful comparison between different formulations of complexity across different fields of study.*
3. *To lay the foundations for formalisations of complexity, in different circumstances.*
4. *To aid the formulation of deeper insights into possible causes of complexity.*
5. *To allow the development of systematic approaches to simplification.*

Additionally I have focused on a particular measure of complexity to capture the difficulty of decomposing expressions and sketched a possible formal structure to relate different formal languages to the analysis of the sources of complexity in specific systems, which also provides a framework for the study of systematic methods simplification. Ancillary work has included a short survey of the vast number of formulations of complexity (section 8 on page 136) and a software tool for the exploration of syntactic structures (described in section 11 on page 188).

### 7.1 Further Work

This work is foundational in that it could form part of a new “science of complexity” [89]. Much more can be done in this regard, including:

- Further investigation into the relation between syntactic structures and the complexity of expressions.
- The development of automatic, or semi-automatic methods of simplification.

- The integration of this work into other's work in the field of "complex systems", especially with regard to identifying possible *causes* of complexity.
- The further development of measures of complexity for different purposes.
- The use of this work as the synthesis of many formulations of complexity, including their categorisation and generalisation from similar families.

Perhaps the most important way this work could be extended is towards a model of the process of modelling itself. This would involve extending and formalising the semantic picture of modelling presented in section 2.3 on page 33, and could be seen as an extension of measure theory to non-numeric structures. As Badii and Politi put it in their conclusion to their book on complexity:

*"The natural extension of the study of complexity... seems, therefore, to point inevitably to a theory of model inference." [36] p.280*

## 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:  $D_s(x) = \min \{T(p) \mid |p| - |p^*| < s \wedge U(p) = x\}$ , 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  $v(G) = m - n + p$ , 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:

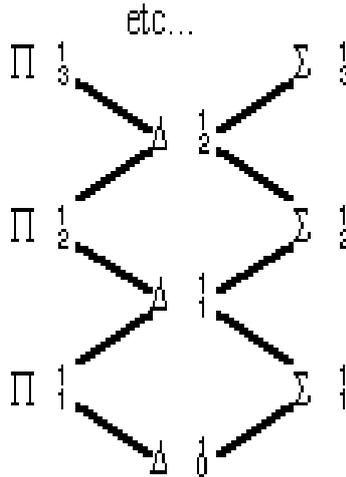
$$\forall x_1 \exists x_2 \forall x_3 \exists x_4 \dots Q x_n R [x_1, \dots, x_n]$$

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

Likewise if the statement is of the form:

$\exists x_1 \forall x_2 \exists x_3 \forall x_4 \dots Qx_n R[x_1, \dots, x_n]$   
 then it is in the class  $\Sigma_n^1$ .

If a statement is provably equivalent to both a  $\Sigma_n^1$  statement and a  $\Pi_n^1$  statement it is called a  $\Delta_n^1$  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  $x+1$ ),  $x \times y$  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 led 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 into 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 ( $\theta: M_f \rightarrow Z$  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)  $\theta(f_1 \oplus \dots \oplus f_n) = \max \{ \theta(f_i) \mid i= 1, \dots, n \}$
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  $\gg$  from  $f_1$  to  $f_2$ , then

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

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

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 than 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 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 been 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 than 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 Landauer-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

$$[(p \rightarrow q) \rightarrow r] \rightarrow [(r \rightarrow p) \rightarrow (s \rightarrow p)]$$

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:

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

and

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

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  $a \rightarrow b$  has a catastrophically simplifying effect on both proofs and models compared to that of  $a \rightarrow a$ .

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  $p(m_1), p(m_2), \dots$  are the probabilities of receiving the messages  $m_1, m_2, \dots$  then the information carried by the message  $n_1, n_2, \dots$  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 than 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 string's 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  $\min \{ |P| \mid \exists D (P, D) \text{ is a description of } S \text{ and } |P| + |D| \leq H(S) + c \}$ , 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  $\epsilon$ -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  $\bar{\alpha} = \alpha_1 \alpha_2 \dots \alpha_n$  (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  $\bar{\alpha} = \alpha_1 \alpha_2 \dots \alpha_n$  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

$\bar{\alpha} = \alpha_1 \alpha_2 \dots \alpha_m$  and  $\bar{\beta} = \beta_{m+1} \beta_{m+2} \dots \beta_n$  are trajectories then

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

4.  $p(\bar{\alpha}) = q(\bar{\alpha}) \Rightarrow f(\{p(\bar{\alpha}), q(\bar{\alpha})\}) = \bar{0}$  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].

## 9 Appendix 2 - Longer Proofs

### 9.1 (Non-existence of) Complexity Measures on Strings

#### Definitions

Let  $L$  be the set of non-empty strings with symbols from  $S = \{a, b, c, \dots\}$ , i.e.  $L = \{a, aa, ab, b, \dots\}$ .

If  $x$  and  $y$  are the same pattern, but using different symbols, I will write  $x \approx y$ , e.g.  $aab \approx bbc$ .

Let  $C:L \rightarrow \mathfrak{R}^+$  be a function to the non-negative reals (this is the intended complexity function).

I will write the concatenation of  $x$  and  $y$  as  $xy$ .

Let  $\wp(x)$  be the set of non-empty substrings of  $x$  (including itself), e.g.  $\wp(aba) = \{a, b, ab, ba, aba\}$ .

I will show the substitution of  $y$  for  $c \in S$  throughout  $x$  as  $x(c/y)$ , e.g.  $aba(a/cd) = cdbcd$ .

Two patterns are irrelevant to each other,  $I(x, y)$ , if they have no symbols in common, i.e.  $I(x, y) \equiv \wp(x) \cap \wp(y) = \emptyset$ , e.g.  $I(aba, cc)$  but not  $I(babcb, ddad)$ .

When  $x$  and  $y$  are irrelevant to each other I will denote their concatenation:  $x \cdot y$ .

Let  $x^n$  be shorthand for the string  $x$  concatenated with itself  $n-1$  times, e.g.  $(ab)^3 = ababab$ .

In all of the below  $w, x, y, z \in L$ ;  $a, b, r, s, t \in S$ ; and  $\alpha, i, n, m, s_1, s_2, \dots, t_1, t_2, \dots \in N$  (where  $N$  is the set of natural numbers).

**Theorem**

If  $S$  has at least three distinct symbols then there is no non-trivial measure from  $\langle L, \cdot, \leq \rangle$  into  $\langle \mathfrak{R}^+, +, \leq \rangle$ , that is, if there is a function  $C:L \rightarrow \mathfrak{R}^+$ , defined on  $L$  to the non-negative real numbers such that,  $\forall x,y \in L$ :

$$C(x \cdot y) = C(x) + C(y) \quad (\text{Irrelevant Join})$$

$$x \leq y \Leftrightarrow C(x) \leq C(y) \quad (\text{Subform})$$

such that:

$$x \in \wp(y) \Rightarrow x \leq y \quad (\text{Subpattern Property})$$

$$I(x, y), s \in \wp(x) \Rightarrow C(x(s/y)) = C(x) + C(y) \quad (\text{Irrelevant Substitution})$$

Then for all  $x$ ,  $C(x) = 0$ .

There needs to be at least 3 symbols for the technical reason that sometimes you sometimes temporarily need a third symbol to apply “Irrelevant Substitution”. For all of the below I use a constant,  $k \geq 0$ , defined as  $C(ss)$ . The triviality of the measure will come from a proof that  $k = 0$ .

Proof:

**Lemma 1**

$$C(s) = 0$$

Proof of Lemma:

Choose  $t \in S$ ,  $s \neq t$ , now

$$C(t) = C(t(t/s)) = C(t) + C(s) .$$

**Lemma 2**

If  $x$  has no repetitions of any symbols in it, then  $C(x) = 0$ .

Proof of Lemma:

Let  $x = s_1s_2\dots s_n$ , where  $s_i \in S$  are pairwise distinct.

Now using repeated instances of Irrelevant Join and Lemma 1 we have:

$$C(s_1s_2\dots s_n) = C(s_1) + C(s_2\dots s_n) = \dots = C(s_1) + \dots + C(s_n) = 0 .$$

**Lemma 3:**

If  $x \approx y$ , then  $C(x) = C(y)$ . (The *Substitution of Symbols*)

Proof of Lemma:

Let  $s_1, s_2, \dots, s_n$  be the symbols that occur in  $x$  and  $t_1, t_2, \dots, t_n$  be the corresponding symbols in  $y$ .

Now using Lemma 1 and repeated applications of Irrelevant Substitution we have:

$$C(x) = C(x) + C(t_1) + \dots + C(t_n) = C(x(s_1/t_1) \dots (s_n/t_n)) = C(y).$$

**Lemma 4:**

$$C(s^{n^m}) = mC(s^n).$$

Proof of Lemma:

By induction on  $m$ .

Base step:  $C(s^{n^1}) = C(s^n) = 1.C(s^n)$ .

Inductive step:

$$\begin{aligned} C(s^{n^{m+1}}) &= C(s^{n^m}) = C(t^{n^m}(t/s^n)) = C(t^{n^m}) + C(s^n) = C(s^{n^m}) + C(s^n) \\ &= (mC(s^n) + C(s^n)) = (m+1)C(s^n). \end{aligned}$$

**Lemma 5:**

$$C(s^{(2^m)}) = mk$$

Proof of Lemma:

This is a special case of Lemma 4, with  $n=2$ .

**Lemma 6:**

$$C(s^n) = k \log_2 n.$$

Proof of Lemma:

now given  $m \geq 1$ , choose  $\alpha(m) > 0$  such that  $2^\alpha \leq n^m \leq 2^{\alpha+1}$ .

So  $a^{2^\alpha}$  is a substring of  $a^{n^m}$  which is a substring of  $a^{2^{\alpha+1}}$ .

By the *subpattern* property, we have

$$C(a^{2^\alpha}) \leq C(a^{n^m}) \leq C(a^{2^{\alpha+1}}),$$

therefore by Lemmas 4 and 5:  $k\alpha \leq mC(a^n) \leq k(\alpha+1)$ ,

so  $k\frac{\alpha}{m} \leq C\left(a^n\right) \leq k\frac{\alpha+1}{m}$  since  $m > 0$ .

But by the choice of  $\alpha$

$$\alpha \leq m \log_2 n \leq \alpha + 1$$

so  $k\frac{\alpha}{m} \leq k \log_2 n \leq k\frac{\alpha+1}{m}$ , as  $m > 0$ .

Given that  $m$  can be arbitrarily large,  $C\left(a^n\right)$  must be arbitrarily close to  $k \log_2 n$ .

**Lemma 7:**

$$C\left(x^n\right) = k \log_2 n + C(x).$$

Proof of Lemma:

Using Lemma 6, we have:

$$C\left(x^n\right) = C\left(s^n (s/x)\right) = C\left(s^n\right) + C(x) = k \log_2 n + C(x).$$

**Lemma 8:**

If  $x$  is a rotation of  $y$  then  $C(x) = C(y)$

(where  $x$  is a rotation of  $y$  if  $\exists w, z \in L((wz = x) \wedge (zw = y))$ ).

Proof of Lemma:

let  $w$  and  $z$  be strings such that  $x = wz, y = zw$ .

For  $n > 1$ ,  $(wz)^{n-1}$  is a sub-pattern of  $(zw)^n$  which is a sub-pattern of  $(wz)^{n+1}$ ,

so by the sub-pattern property

$$\begin{aligned} C\left((wz)^{n-1}\right) &\leq C\left((zw)^n\right) \leq C\left((wz)^{n+1}\right) \\ \Rightarrow k \log_2 (n-1) + C(wz) &\leq k \log_2 n + C(zw) \leq k \log_2 (n+1) + C(wz) \end{aligned}$$

(using lemma 7)

$$\Rightarrow k \log_2 \left(\frac{n-1}{n}\right) + C(wz) \leq C(zw) \leq k \log_2 \left(\frac{n+1}{n}\right) + C(wz)$$

Since  $n$  can be arbitrarily large,  $C(wz) = C(zw)$ .

Now to show that  $k = 0$ .

$$\begin{aligned}
 & k + k\log_2 3 \\
 &= k + C(c^3) && \text{Lemma 6} \\
 &= k + C(c^3) + C(a) + C(b) && \text{Lemma 1} \\
 &= k + C(c^3) + C(ab) && \text{Irrelevant Join} \\
 &= k + C(c^3(c/(ab))) && \text{Irrelevant Substitution} \\
 &= k + C(ababab) && \text{Substitution expanded} \\
 &\geq k + C(ababa) && \text{Sub-pattern} \\
 &= k + C(abaab) && \text{Lemma 8} \\
 &= C(c^2) + C(abaab) && \text{Definition of } k \\
 &= C(c^2(c/(abaab))) && \text{Irrelevant Substitution} \\
 &= C(abaababaab) && \text{Substitution expanded} \\
 &\geq C(abaababaa) && \text{Sub-pattern} \\
 &= C(baaabaaba) && \text{Lemma 8} \\
 &\geq C(baaabaab) && \text{Sub-pattern} \\
 &= C(aabbbaab) && \text{Lemma 8} \\
 &\geq C(aabbbaa) && \text{Sub-pattern} \\
 &= C(aaaaabb) && \text{Lemma 8} \\
 &= C(a^5) + C(b^2) && \text{Irrelevant Join} \\
 &= k\log_2 5 + k. && \text{By the definition of } k \text{ and Lemma 7}
 \end{aligned}$$

Finally I will show that  $\forall x \in L; C(x) = 0$ , which will prove the theorem.

Let  $x$  be a string containing the following  $n$  pairwise distinct symbols:  $s_1, \dots, s_n$ .

Construct a new string  $\hat{x} = f_n(f_{n-1}(\dots f_1(x)\dots))$  from  $x$ , using  $n$  applications of the following:  $f_i(x)$  is constructed from  $f_{i-1}(x)$  (using the convention that  $f_0(x)$  is  $x$ ) by replacing all instances of  $s_i$  in  $f_{i-1}(x)$  that were also in  $x$  using these four steps:

- (1) rotate  $x$  until the first symbol is the chosen instance of  $s_i$ ;
- (2) append the string  $s_1 \dots s_{i-1}$  to the front of the rotated string;

- (3) rotate the new string until the chosen instance of  $s_i$  is at the end;
- (4) append the string  $s_{i+1} \dots s_n$  at the end.

*Fact 1:*

$C(f_i(x)) \geq C(f_{i-1}(x))$  since a rotation does not change a string's complexity by Lemma 7 and appending symbols at the beginning or end of a string can only increase its complexity due to the subpattern assumption.

*Fact 2:*

By construction  $\hat{x} = (s_1 \dots s_n)^{|x|}$ , since by construction every symbol in  $x$  is replaced by the string  $s_1 \dots s_n$ .

By repeated applications of fact 1, we have  $C(\hat{x}) \geq C(x)$ .

So  $C(x) \leq C(\hat{x}) \leq C\left((s_1 \dots s_n)^{|x|}\right) = k \log_2 |x| + C(s_1 \dots s_n) = 0$ , since  $k = 0$  and  $s_1 \dots s_n$  contains no symbol repetitions (Lemma 2).

.....

## 9.2 Cyclomatic Number as a Lower Bound for Minimal Damage Cut

The cyclomatic number can be calculated as the number of arcs minus the number of nodes plus the number of disjoint partitions the graph is in. Each cut of the formula that incurs "damage" removes at least one arc and the formula is not fully "analysed" into components until you are left with a collection of trees and the cyclomatic number of a collection of trees is zero.

## 9.3 Decomposition of Formulas into Complexes

The proof deferred from section 5.3.2 on page 95.

Any formula can be decomposed into complexes. i.e.

For any  $x \in L$  there are  $a_0, \dots, a_n \in \mathbf{Cp}$  and  $c_1, \dots, c_n \in X_0$  such that

$$x = a_0^{c_1/a_1} \dots^{c_n/a_n}$$

and for  $i \leq n$ ,  $\neg R(a_0^{c_1/a_1} \dots^{c_i/a_i}, c_i)$  so that

$$C(x) = C(a_0) + \dots + C(c_n).$$

Note that this is not necessarily a unique decomposition.

Proof:

By induction on length of  $x$ :

Base Step

$x$  is of length 1  $\Rightarrow x \in X_0 \Rightarrow x \in \mathbf{Cp}$

and thus is its own decomposition.

Induction Step

$x \in \mathbf{Cp} \Rightarrow$  trivial decomposition of itself.

$x \notin \mathbf{Cp} \Rightarrow \exists y \in P(x) - \{x\} - X_0; \neg R(x^y/c, y)$  Defn **Cp**

Now by the induction hypothesis  $x^y/c$  and  $y$  have decompositions:

$$x^y/c = a_0^{c_1}/a_1 \dots c_n/a_n$$

$$y = b_0^{k_1}/b_1 \dots k_m/b_m$$

These two decompositions are irrelevant to each other, so

$$\begin{aligned} x &= (x^y/c)^c/y \\ &= a_0^{c_1}/a_1 \dots c_n/a_n^c/b_0^{k_1}/b_1 \dots k_m/b_m \quad \text{as } \neg R(a_0^{c_1}/a_1 \dots c_n/a_n, b_0^{k_1}/b_1 \dots k_m/b_m) \end{aligned}$$

Now  $\forall i < n \neg R(a_0^{c_1}/a_1 \dots c_i/a_i, a_{i+1}), \neg R(a_0^{c_1}/a_1 \dots c_n/a_n, b_0)$

and for  $\forall j < m \neg R(a_0^{c_1}/a_1 \dots c_n/a_n^c/b_0^{k_1}/b_1 \dots k_j/b_j, b_{j+1}),$  since  $\neg R(x^y/c, y)$ .

Now, of course,

$$\begin{aligned} C(x) &= C(a_0) + \dots + C(a_n) + \\ &\quad C(b_0) + \dots + C(b_m). \end{aligned}$$

.....

### 9.4 Generating a Measure from a Function on the Complexes

The proof from section 5.3.2 on page 95. First a lemma I will use in the proof:

**Lemma**

$$y, z \in \wp(x), c, k \in X_0 - \wp(x), \neg R(x^y/c, y), \neg R(x^z/k, z)$$

$$\Rightarrow \neg R(y, z) \text{ or } y \in \wp(z) \text{ or } z \in \wp(y)$$

Proof:

By induction on the depth of formulas.

Base (depth 0):

$$\text{depth}(x)=0 \Rightarrow x \in X_0 \Rightarrow \wp(x)=\{x\}$$

$$y, z \in \wp(x) \Rightarrow y=x=z \Rightarrow y \in \wp(z)$$

Induction step:

Assume lemma is true for all formulas of depth less than  $x$  ( $\text{depth}(x) > 0$ ).

One of:

$$(i) \quad x = uw$$

$$(ii) \quad x = bst$$

For some  $w, s, t \in L$ ,  $u \in X_1$ ,  $b \in X_2$ ,  $\text{depth}(w)$ ,  $\text{depth}(s)$ ,  $\text{depth}(t) < \text{depth}(x)$

*Case (i):*  $x = uw$

$$y, z \in \wp(uw) = \{uw\} \cup \wp(w)$$

There are essentially three possibilities: either (a) both  $y$  and  $z$  equal  $uw$ , (b) only one of them or (c) neither. Take each case in turn:

$$(a) \quad y=uw=z \Rightarrow y \in \wp(z)$$

$$(b) \quad (\text{w.l.o.g.}) \quad y=uw, z \in \wp(w) \Rightarrow z \in \wp(uw) = \wp(y)$$

$$(c) \quad y, z \in \wp(w)$$

by the inductive hypothesis since  $\neg R(x^y/c, y)$ ,  $\neg R(x^z/k, z) \Rightarrow \neg R(w^y/c, y)$ ,  $\neg R(w^z/k, z)$  one of  $\neg R(y, z)$ ,  $y \in \wp(z)$  or  $z \in \wp(y)$  holds.

*Case (ii):*  $x = bst$

$$y, z \in \wp(bst) = \{bst\} \cup \wp(s) \cup \wp(t)$$

Here there are essentially four possibilities (modulus the obvious symmetries of  $s$  and  $t$ ): either (a) both  $y$  and  $z$  equal  $bst$ , (b) one is equal to  $bst$  and the other is a member

of  $\wp(s)$ , (c) both are members of  $\wp(s)$  or (d) one is a member of  $\wp(s)$  and the other a member of  $\wp(t)$ . Taking each possibility in turn:

$$(a) \quad y=bst=z \Rightarrow y \in \wp(z)$$

$$(b) \quad (\text{w.l.o.g.}) \ y=bst, \ z \in \wp(s) \Rightarrow z \in \wp(bst) = \wp(y)$$

$$(c) \quad (\text{w.l.o.g.}) \ y, z \in \wp(s)$$

$$\neg R(x^y/c, y), \neg R(x^z/k, z) \Rightarrow \neg R(s^y/c, y), \neg R(s^z/k, z)$$

and so by the inductive hypothesis one of  $\neg R(y, z)$ ,  $y \in \wp(z)$  or  $z \in \wp(y)$  holds.

$$(d) \quad (\text{w.l.o.g.}) \ y \in \wp(s) - \wp(t), \ z \in \wp(t) - \wp(s)$$

$$\neg R(x^y/c, y)$$

$$\Rightarrow \neg R((bst)^y/c, y)$$

as  $x=bst$

$$\Rightarrow \neg R(b(s^y/c)t, y)$$

$y \notin \wp(t)$

$$\Rightarrow \wp(b(s^y/c)t) \cap \wp(y) = \emptyset$$

Defn R

$$\Rightarrow [\{b(s^y/c)t\} \cup \wp(s^y/c) \cup \wp(t)] \cap \wp(y) = \emptyset$$

$$\Rightarrow \wp(t) \cap \wp(y) = \emptyset$$

Similarly

$$\neg R(x^z/k, z) \Rightarrow \wp(s) \cap \wp(z) = \emptyset$$

Now if  $q \in \wp(x)$  then either:  $q=bst$ ,  $q \in \wp(s)$  or  $q \in \wp(t)$ . If  $q=bst$  then  $q \notin \wp(y) \cap \wp(z)$  by the assumptions of (d). If  $q \in \wp(s)$  then  $q \notin \wp(z)$  since  $\wp(s) \cap \wp(z) = \emptyset$ . If  $q \in \wp(t)$  then  $q \notin \wp(y)$  since  $\wp(t) \cap \wp(y) = \emptyset$ . In any of these cases  $q \notin \wp(y) \cap \wp(z)$  so we have  $\wp(y) \cap \wp(z) = \emptyset$ , i.e.  $\neg R(y, z)$ .

This completes the proof of possibility (d), hence cases (i) and (ii) and thus the induction step.

.....

Now for the main proof.

A function,  $g$ , on  $\mathbf{Cp}$  in  $L$ ,  $g: \mathbf{Cp} \rightarrow \mathfrak{R}^+$ ,

$$x \in X_0 \Rightarrow g(x) = 0$$

$$x \approx y \Rightarrow g(x) = g(y)$$

$$x \leq y \Rightarrow g(x) \leq g(y).$$

will generate a unique complexity measure,  $C: L \rightarrow \mathfrak{R}^+$ , on  $L$ , thus:

$$C(x) \equiv_{\text{def}} g(a_0) + g(a_1) + g(a_2) + \dots + g(a_n),$$

for some decomposition:  $x = a_0^{c_1}/a_1^{c_2}/a_2 \dots^{c_n}/a_n$ , into complexes:  $a_0, a_1, a_2, \dots, a_n \in \mathbf{Cp}$ , as above and hence an ordering,  $\leq$ , on  $L$ :

$$x \leq y \Leftrightarrow C(x) \leq C(y).$$

Proof:

We have to check two things:

That  $C$  is well-defined, that is given two decompositions of  $x$  the sum of  $g$  over these is the same, that is:

$$(1) x = a_0^{c_1}/a_1 \dots^{c_n}/a_n = b_0^{k_1}/b_1 \dots^{k_m}/b_m$$

$$\Rightarrow g(a_0) + g(a_1) + \dots + g(a_n) = g(b_0) + g(b_1) + \dots + g(b_m).$$

And that the complexity of the result of an irrelevant substitution is the sum of the complexity of the formula that is substituted in and the formula it is substituted into:

$$(2) k \in X_0 \cap \wp(x), \neg R(x, y)$$

$$\Rightarrow C(x^k/y) = C(x) + C(y)$$

I do this by induction on the maximum size of formulas.

Base (size 1)

$$|x|=1 \Rightarrow x \in X_0 \Rightarrow C(x) = 0 = g(x)$$

and so  $C$  is well defined, showing (1).

$$k \in X_0 \cap \wp(x)$$

$$\Rightarrow k = x$$

$$\Rightarrow x^k/y = y$$

$$\Rightarrow C(x^k/y) = C(y) = C(y) + 0 = C(y) + g(x) = C(y) + C(x)$$

showing (2).

Induction Step

Assume (1) and (2) hold for all formulas with size less than that of  $x$ .

Suppose that  $x$  can be decomposed by two different non-trivial irrelevant substitutions. So that:

$$\exists y, z \in \wp(x) - \{x\}, \exists c, k \in X_0 - \wp(x), \neg R(x^y/c), \neg R(x^z/k, z)$$

If we could show that  $C(x^y/c) + C(y) = C(x^z/k) + C(z)$ , then by the induction hypothesis  $C$  would be a well defined measure on  $x^y/c$ ,  $y$ ,  $x^z/k$  and  $z$  and thus we would have shown (1) for all formulas up to and including size  $|x|$ .

Now if  $k \in \wp(x) \cap X_0$ ,  $\neg R(x, y)$ ,  $x$  and  $y$  have decompositions into complexes:

$$x = a_0^{c_1}/a_1 \dots a_n^{c_n}/a_n, \forall i \leq n \neg R(a_0^{c_1}/a_1 \dots a_i^{c_i}/a_i, a_{i+1})$$

$$y = b_0^{k_1}/b_1 \dots b_m^{k_m}/b_m, \forall j \leq m \neg R(b_0^{k_1}/b_1 \dots b_j^{k_j}/b_j, b_{j+1})$$

$$\Rightarrow C(x^k/y)$$

$$= C(a_0^{c_1}/a_1 \dots a_n^{c_n}/a_n (k/(b_0^{k_1}/b_1 \dots b_m^{k_m}/b_m)))$$

$$= C(a_0^{c_1}/a_1 \dots a_i^{c_i}/(a_i^{k/(b_0^{k_1}/b_1 \dots b_m^{k_m}/b_m)})) \dots a_n^{c_n}/a_n) \quad \text{as for one } i \leq n, k \in a_i$$

$$= C(a_0^{c_1}/a_1 \dots a_i^{c_i}/a_i^{k/(b_0^{k_1}/b_1 \dots b_m^{k_m}/b_m)} \dots a_n^{c_n}/a_n) \quad \text{as } k \text{ only occurs in } a_i$$

$$= C(a_0^{c_1}/a_1 \dots a_i^{c_i}/a_i^{k/b_0^{k_1}/b_1 \dots b_m^{k_m}/b_m} \dots a_n^{c_n}/a_n) \text{ as } k_1, \dots, k_m \text{ do not occur in } a_0, \dots, a_i$$

$$= g(a_0) + \dots + g(a_i) + g(b_0) + \dots + g(b_m) + g(a_{i+1}) + \dots + g(a_n)$$

since  $C$  is well-defined on  $a_0, \dots, a_n, b_0, \dots, b_m$  by induction hypothesis

$$= g(a_0) + \dots + g(a_n) + g(b_0) + \dots + g(b_m) \quad \text{by rearranging}$$

$$= C(x) + C(y)$$

So all we have left to prove is  $\exists y, z \in \wp(x) - \{x\}, \exists c, k \in X_0 - \wp(x), \neg R(x^y/c), \neg R(x^z/k, z)$

$$\Rightarrow C(x^y/c) + C(y) = C(x^z/k) + C(z) \text{ for formula of size } |x|.$$

Using the Lemma immediately above, either:

(i)  $\neg R(x, y)$

(ii)  $y \in \wp(z)$  or  $z \in \wp(y)$ .

Case (i)

$$\neg R(x^y/c, y) \Rightarrow \neg R(x^z/k^y/c, y) \quad \text{as } \neg R(k, y)$$

$$\Rightarrow C(x^z/k) = C(x^z/k^y/c) + C(y) \quad \text{by induction hypothesis as } |x^z/k| < |x|$$

$$\text{similarly } C(x^y/c) = C(x^y/c^z/k) + C(z)$$

Now  $x$  and  $y$  occur separately in  $x$  so  $x^z/k^y/c = x^y/c^z/k$ , and thus:

$$C(x^y/c) - C(z) = C(x^y/c^z/k) = C(x^z/k^y/c) = C(x^z/k) - C(y)$$

Case (ii)

$$y \in \wp(z) \text{ or } z \in \wp(y).$$

W.l.o.g. say,  $y \in \wp(z)$ .

$$\neg R(x^z/k, z) \Rightarrow \neg R(x^y/c^{(zy/c)/k}, (zy/c)) \quad \text{as } y \in \wp(z), y \notin \wp(x^z/k)$$

$$C(x^y/c) = C(x^y/c^{(zy/c)/k}, (zy/c)) \quad \text{as } x^y/c = x^y/c^{(zy/c)/k}, (zy/c)$$

$$= C(x^y/c^{(zy/c)/k}) + C(zy/c) \quad \text{as } \neg R(x^y/c^{(zy/c)/k}, (zy/c))$$

$$= C(x^z/k) + C(zy/c) \quad \text{as } x^y/c^{(zy/c)/k} = x^z/k$$

$$\neg R(x^y/c, y), zy/c \in \wp(x^y/c) \Rightarrow \neg R(zy/c, y)$$

$$C(z) = C(zy/c^y) = C(zy/c) + C(y) \quad \text{as } \neg R(zy/c, y)$$

$$C(x^y/c) = C(x^z/k) + C(z) - C(y) \quad \text{substituting out } C(zy/c)$$

Which rearranged shows:  $C(x^y/c) + C(y) = C(x^z/k) + C(z)$

.....

## 9.5 Three Conditions that are Equivalent on a Weak Complexity

### Measure

This is the proof from section 5.3.5 on page 104. But first I show that condition (i) implies the subformula property.

#### (i) $\Rightarrow$ the Subformula Property

If  $R(x, y) \Rightarrow C(bxy) > \max\{C(x), C(y)\}$  then

$$y \in P(x) \Rightarrow C(y) \leq C(x)$$

Proof:

The case  $y=x$  is trivial so assume  $y \neq x$ .

By induction on depth of  $x$ :

Base:

$$x \in X_0 \Rightarrow y \in X_0 \Rightarrow C(y)=0=C(x)$$

Induction Step:

$x$  is of the form

$$(1) \text{ } uz$$

or (2)  $bzw$

*Case (a):*  $x=uz$

$$y \in P(z) \text{ as } y \neq x$$

$$\Rightarrow C(y)$$

$$\leq C(z)$$

by induction hypothesis

$$= C(uz) = C(x)$$

*Case (b):*  $x=bzw$

$$\Rightarrow y \in \wp(z) \text{ or } y \in \wp(w), \text{ say w.l.o.g. } y \in \wp(z)$$

$$\text{if } \neg R(z,w) C(x) = C(bzw)$$

$$= C(z) + C(w)$$

by Irrel Join

$$\geq C(y) + C(w)$$

by induction hypothesis

$$\geq C(y)$$

$$\text{if } R(z,w) C(x) = C(bzw)$$

$$> \max\{C(y), C(w)\}$$

by Rel Join

$$\geq C(y).$$

.....

The following three conditions are equivalent on a weak complexity measure, for  $b \in X_2, x, y \in L$ :

- (i)  $R(x, y) \Rightarrow C(bxy) > \max\{C(x), C(y)\}$  (Rel Join)
- (ii)  $y \in \wp(x) - \{x\}, c \in X_0 - (x), R(x^y/c, y) \Rightarrow C(y) < C(x)$  (Rel Subform)
- (iii)  $y \in \wp(x) - \{x\} - X_0, x \in \mathbf{Cp} \Rightarrow C(y) < C(x)$  (Subform of Complex)

Note that I have written them here in terms of the ordering on the reals rather than the ordering  $\leq$  on  $L$ , for convenience. The equivalence is assured due to the homomorphic mapping between them.

Proof: (i)  $\Rightarrow$  (ii)

By induction on the maximum depth of  $x$ .

Assuming  $y \in \wp(x) - \{x\}, c \in \wp X_0 - (x), R(x^y/c, y)$

Base:

$x \in X_0$ , vacuously true as  $\wp(x) - \{x\} = \emptyset$

Induction step:

either (a)  $x$  is form  $(uz)$  for some  $u \in X_1, z \in L$

or (b)  $x$  is of form  $(bwz)$  for some  $b \in X_2, w, z \in L$

Case (a):  $x=uz$

$$c \in X_0 - \wp(x) \Rightarrow c \in X_0 - \wp(y)$$

$$\neg R(uc, y) \Rightarrow \neg R(uy^y/c, y)$$

$$\Rightarrow \neg R(x^y/c, y)$$

$$\Rightarrow y \neq uy$$

otherwise  $R(uy^y/c, y)$

$$\Rightarrow y \neq z$$

$$y \in \wp(z)$$

as  $y \in \wp(uz) = \wp(x)$

$$R(x^y/c, y) \Rightarrow R(uz^y/c, y)$$

$$\Rightarrow R(z^y/c, y)$$

$$\Rightarrow C(y)$$

$$< C(z)$$

by the induction hypothesis

$$= C(uz)$$

$$= C(x).$$

Case (b):  $x=bwz$

If  $R(w, z)$

$$x \neq y \Rightarrow y \in \wp(w) \text{ or } y \in \wp(z), \text{ say w.l.o.g. } y \in \wp(w)$$

$$C(x) = C(bwz)$$

$$> \max\{C(w), C(z)\}$$

from (i).

$$\geq \max\{C(y), C(z)\}$$

as  $y \in P(w) \Rightarrow C(y) \leq C(w)$

$$\geq C(y)$$

If  $\neg R(w, z)$

$$y \neq w \text{ and } y \neq z$$

as, if not, say (w.l.o.g.)  $y=w$  then:

$$\begin{aligned}
 R(x^y/c, y) &\Rightarrow R(bwz^y/c, y) && \text{by assumption Case (b)} \\
 &\Rightarrow R(byz^y/c, y) && \text{by assumption } y=w \\
 &\Rightarrow R(bc(z^y/c), y) \\
 &\Rightarrow R(z^y/c, y) && \text{as } \neg R(y, c) \text{ by choice of } c \\
 &\Rightarrow R(z, y) \\
 &\Rightarrow R(z, w) && \text{as } y=w
 \end{aligned}$$

which is contrary to assumption.

$$\Rightarrow y \in \wp(z) - \{z\} \text{ or } y \in \wp(w) - \{w\}$$

$$R(x^y/c, y) \Rightarrow R(z^y/c, y) \text{ or } R(w^y/c, y), \quad \text{as } \neg R(w, z)$$

say w.l.o.g.  $R(z^y/c, y) [\Rightarrow \neg R(w^y/c, y) \text{ as } \neg R(w, z)]$

If  $y \in (z)$ ,  $y \neq z$  then

$$\begin{aligned}
 C(y) &< C(z) && \text{by induction hypothesis} \\
 &\leq C(x) && \text{as } z \in \wp(x)
 \end{aligned}$$

If  $y \in \wp(w) - \{w\}$

then  $R(w^y/c, y)$

or else if  $\neg R(w^y/c, y)$ :

$$R(x^y/c, y) \Rightarrow R((bwz)^y/c, y)$$

$$\Rightarrow R(b(w^y/c)(z^y/c), y)$$

$$\Rightarrow R(z^y/c, y) \quad \text{as } \neg R(w^y/c, y)$$

$$\Rightarrow R(z, y)$$

$$\Rightarrow R(z, w) \quad \text{as } y \in \wp(w)$$

which is contrary to the *Case (b)* assumption

$$\Rightarrow C(y) < C(w) \quad \text{by the induction hypothesis}$$

$$\leq C(x) \quad \text{as } w \in \wp(x)$$

.....

Proof: (ii)  $\Rightarrow$  (i)

$$R(x,y) \Rightarrow R(bxc,y) \quad \text{for suitably chosen } c \in X_0$$

$$\Rightarrow R((bxy)^y/c,y)$$

$$\text{Now } y \in \wp(bxy) - \{bxy\} \text{ so } C(y) < C(bxy) \quad \text{by (ii)}$$

$$\text{Likewise } C(x) < C(bxy)$$

$$\Rightarrow C(bxy) > \max\{C(y), C(x)\}.$$

.....

Proof: (iii)  $\Rightarrow$  (ii)

$$y \in \wp(x) - \{x\}, c \in X_0 - \wp(x), R(x^y/c,y)$$

By induction on the length of the decomposition of  $x$ .

*Base:*

$$x \in \mathbf{Cp} \Rightarrow C(y) < C(x) \quad \text{by (iii)}$$

*Induction Step:*

$$x \notin \mathbf{Cp} \Rightarrow \text{there is a } z \in \wp(x) - \{x\} - X_0, z \in \mathbf{Cp}; \neg R(x^z/k,z) \quad k \in X_0 - \wp(x)$$

$$\Rightarrow y \neq z \quad \text{as } R(x^y/c,y)$$

$$\Rightarrow C(x) = C(x^z/k) + C(z) \quad \text{by Irrel Subs}$$

$$(*) \quad \Rightarrow C(x^z/k) \leq C(x) \text{ and } C(z) \leq C(x)$$

$$y \in \wp(x), \neg R(x^z/k,z) \text{ means either}$$

$$(i) y \in \wp(x^z/k)$$

$$\text{or } (ii) y \in \wp(z)$$

*Case (i):*

$$\text{if } y = x^z/k \text{ then } x^z/k \in \wp(x) \text{ but } \neg R(k,x) \text{ as } k \in X_0 - \wp(x)$$

$$\text{so } y \neq x^z/k.$$

$$R(x^y/c,y) \Rightarrow R(x^z/k^y/c,y) \quad \text{as } \neg R(x^z/k,z) \text{ and } y \in \wp(x^z/k)$$

$$\Rightarrow C(y) < C(x^z/k) \quad \text{by induction hypothesis}$$

$$\leq C(x) \quad \text{from } (*)$$

Case (ii):

$z \in \mathbf{Cp}$

$$\Rightarrow C(y) < C(z)$$

by (iii)

$$\leq C(x)$$

from (\*).

.....

Proof: (ii)  $\Rightarrow$  (iii)

$y \in \wp(x) - \{x\}, x \in \mathbf{Cp}$

$$x \notin X_0$$

as  $\wp(x) - \{x\} \neq \emptyset$

$$y \notin X_0$$

$$\Rightarrow R(x^y/c, y)$$

by Defn **Cp**

$$\Rightarrow C(y) < C(x)$$

by (ii).

.....

## 10 Appendix 3 - Formalisation of Syntactic Structure

### 10.1 Formalisation

In all of the below there is a set of atomic symbols  $A = \{a_1, a_2, \dots, a_n\}$  and a sufficient supply of variable symbols,  $V = \{v_1, v_2, \dots\}$  that are distinct from each other and  $A$ . The brackets in the below are punctuation.

#### 10.1.1 The syntax of trees

Let  $L$  be a set, then the set of *finite ordered finitely branching trees labelled from  $L$*  (or *ordered trees* for short),  $T_L$ , is built up recursively:

O1: if  $l \in L$  then  $(l, \emptyset) \in T_L$ ;

O2: if  $l \in L$  and  $b_1, \dots, b_n \in T_L$  then  $(l, (b_1, \dots, b_n)) \in T_L$ .

Given a set of variables,  $V$ , that are distinct from other symbols the set of *variable trees based on  $L$  and  $V$* ,  $T_L^+$ , is built up as follows:

V1: if  $t \in L \cup V$  then  $(t, \emptyset) \in T_L^+$ ;

V2: if  $l \in L$  and  $b_1, \dots, b_n \in T_L^+ \cup V$  then  $(l, (b_1, \dots, b_n)) \in T_L^+$ .

Given a variable tree,  $t$ , the variables,  $v(t)$ , can be re-collected as follows:

C1: if  $t = (v, \emptyset)$  and  $v \in V$  then  $v(t) = \{v\}$ ;

C2: if  $t = (l, (b_1, \dots, b_n))$  then  $v(t) = v((l, \emptyset)) \cup \bigcup_{i=1, \dots, n} v(b_i)$ .

#### 10.1.2 The syntax of rules

Given a set of syntaxes  $SS = S_1, \dots, S_n$ , the *rules*,  $R_{SS}$ , are built up as follows:

R1:  $(\Rightarrow) \in R_{SS}$ ;

R2: if  $(a_1, \dots, a_n \Rightarrow) \in R$  and  $x \in T_A^+$  then  $(a_1, \dots, a_n, x \Rightarrow) \in R_{SS}$ ;

R3: if  $(a_1, \dots, a_n \Rightarrow) \in R$ ,  $x \in T_A^+$  and  $\{v_1, \dots, v_n\} = v(x) \cup \bigcup_{i=1, \dots, n} v(a_i)$  then  $(a_1, \dots, a_n \Rightarrow, x, v_1 \in S_1, \dots, v_n \in S_n) \in R_{SS}$ .

#### 10.1.3 The syntax of syntactic structures:

The set of *syntactic structures*,  $S$ , is built up as follows:

S1:  $(\emptyset, \emptyset) \in S$ ;

S2: if  $\{S_1, \dots, S_n\} \subseteq S$  then  $(\{S_1, \dots, S_n\}, \emptyset) \in S$ ;

S3: if  $SS = \{S_1, \dots, S_n\}$ ,  $(SS, \emptyset) \in S$ , and  $\{r_1, \dots, r_k\} \subseteq R_{SS}$  then  $(SS, \{r_1, \dots, r_k\}) \in S$ .

If  $(SS, R)$  is a syntactic structure then  $SS$  is its *sub-syntaxes* and  $R$  is its *rules*. A syntax is *dependant* on a second if either this second syntactic structure is a sub-syntax of it or it is a sub-syntax of another syntactic structure that it is dependent on. Note that this definition includes *only* those syntactic structures that can be built up recursively like this, so there are no 'circular' structures with syntactic structures being dependent on themselves. If you draw a graph with syntactic structures as nodes and the relation of *being a sub-syntax of* being a directed arc then any collection of syntactic structures forms an acyclic digraph.

#### 10.1.4 Generation from syntactic structures

Each syntactic structure,  $S=(SS, R)$ , generates a set of trees,  $Gen(S)$ , as follows:

G1: if  $(\Rightarrow x, v_1 \in S_1, \dots, v_n \in S_n) \in R$  and  $t_1 \in Gen(S_1), \dots, t_n \in Gen(S_n)$  then  $(\Rightarrow x, v_1 \in S_1, \dots, v_n \in S_n)[t_1, \dots, t_n] \in Gen(S)$ ;

G2: if  $(a_1, \dots, a_m \Rightarrow x, v_1 \in S_1, \dots, v_n \in S_n) \in R$ ,  $t_1 \in Gen(S_1), \dots, t_n \in Gen(S)$  such that  $a_1(v_1/t_1) \dots (v_n/t_n) \in Gen(S)$ ,  $\dots$ ,  $a_m(v_1/t_1) \dots (v_n/t_n) \in Gen(S)$  and  $s_1, \dots, s_k$  are the  $t_1, \dots, t_n$  such that the corresponding  $v_1, \dots, v_n$  do not occur as subtrees of any of  $a_1, \dots, a_m$  then  $(a_1, \dots, a_m \Rightarrow x, v_1 \in S_1, \dots, v_n \in S_n)[a_1(v_1/t_1) \dots (v_n/t_n), \dots, a_m(v_1/t_1) \dots (v_n/t_n), s_1, \dots, s_k] \in Gen(S)$ ;

where  $A[b_1, \dots, b_n]$  is the tree with  $A$  as the top node label and  $b_1, \dots, b_n$  are the branches and  $a(x/y)$  is the tree  $a$  with every subtree  $x$  replaced by subtree  $y$ .

Thus the trees *generated* by a syntactic structure like this are trees with rules as node labels and branches of the antecedents and other substitutions to be used in an application of the rule.

#### 10.1.5 Production from trees

Each tree,  $t \in Gen(S)$ , *produces* a tree,  $Prod(t)$ , by recursively applying the rules at each node to the *production* of its branch trees and the substitutions from the generation of lower syntactic structures, thus:

(where in the below  $t = (a_1, \dots, a_m \Rightarrow x, v_1 \in S_1, \dots, v_n \in S_n)[s_1, \dots, s_m, t_1, \dots, t_n]$ )

P1: if  $t = (\Rightarrow x, v_1 \in S_1, \dots, v_n \in S_n)[t_1, \dots, t_n]$  then  $Prod(t) = x(v_1/t_1) \dots (v_n/t_n)$ ;

P2: if  $t = (a_1, \dots, a_m \Rightarrow x, v_1 \in S_1, \dots, v_n \in S_n)[s_1, \dots, s_m, t_1, \dots, t_k]$  and  $s_1 = a_1(v_1/r_1) \dots (v_n/r_n), \dots, s_m = a_m(v_1/r_1) \dots (v_n/r_n)$  where  $\{t_1, \dots, t_k\} \subseteq \{r_1, \dots, r_n\}$  then  $Prod(t) = x(v_1/r_1) \dots (v_n/r_n)$ .

### 10.1.6 Complete production

When a the process of *producing* from a tree,  $t$ , extends recursively downwards to its substitutions, we get a *complete production*,  $ComProd(t)$ , thus:

CP1: if  $t = (\Rightarrow x)[ ]$  then  $ComProd(t) = x$ ;

CP2: if  $t = (\Rightarrow x, v_1 \in S_1, \dots, v_n \in S_n)[t_1, \dots, t_n]$  and  $c_1 = ComProd(t_1), \dots, c_n = ComProd(t_n)$  then  $ComProd(t) = x(v_1/c_1) \dots (v_n/c_n)$ ;

CP3: if  $t = (a_1, \dots, a_m \Rightarrow x, v_1 \in S_1, \dots, v_n \in S_n)[s_1, \dots, s_m, t_1, \dots, t_k]$ ,  $s_1 = a_1(v_1/r_1) \dots (v_n/r_n), \dots, s_m = a_m(v_1/r_1) \dots (v_n/r_n)$  where  $\{t_1, \dots, t_k\} \subseteq \{r_1, \dots, r_n\}$  and  $c_1 = ComProd(r_1), \dots, c_n = ComProd(r_n)$  then  $ComProd(t) = x(v_1/c_1) \dots (v_n/c_n)$ .

A complete production of a tree has only symbols as the labels of its nodes and not rules.

### 10.1.7 Complete productive generation from syntactic structures

Each syntactic structure,  $SS$ , can recursively generate the productions of all trees in  $Gen(SS)$ , called the *complete productive generation*,  $CPG(SS)$ , thus:

CPG: if  $t \in Gen(SS)$  then  $ComProd(t) \in CPG(SS)$ .

It is  $CPG(SS)$  that we usually associate with a syntax in normal logical parlance.

## 10.2 The Expressivity of Syntactic Structures

For any phrase structured grammar (PSG), there is a  $SS$  whose complete productive generation is the language generated by the PSG.

### Proof Outline:

Let the PSG be defined in the usual manner by: a starting symbol,  $S$ ; a set of variables,  $V = \{S, A, B, C, \dots\}$ ; a set of terminal symbols,  $T = \{a, b, c, \dots\}$ ; and a set

of production rules, P, of the form  $g_i \rightarrow \bar{g}_i$  where  $g_i, \bar{g}_i \in (V \cup T)^*$ , where  $L^*$  indicates the set of finite sequences made from the set  $L$ .

Basically I simulate the sequences of symbols with equivalent trees plus a count of how many non-terminals are left and then extract these to another syntactic level. Then the production rules of the PSG translate across in a straight forward manner.

Define a SS,  $S_1 = (\emptyset, R_0)$ , where  $R_0$  includes the following rules:

- PSG1:  $\Rightarrow W(S, s(0))$  - the starting symbol  
 PSG2:  $W(a, p(b, s(c))) \Rightarrow W(a, p(s(b), c))$  - associativity of P  
 PSG3:  $W(a, p(b, 0)) \Rightarrow W(a, b)$  - answer of p  
 PSG4:  $W(a, m(s(b), s(c))) \Rightarrow W(a, m(b, c))$  - associativity of m  
 PSG5:  $W(a, m(b, 0)) \Rightarrow W(a, b)$  - answer of m  
 PSG6:  $W(J(a, J(b, c)), x) \Rightarrow W(J(J(a, b), c), x)$  - associativity of J  
 PSG7:  $W(J(J(a, b), c), x) \Rightarrow W(J(a, J(b, c)), x)$  - associativity of J

The intention is that:  $W(a, n)$  represents a word, a, being processed in the PSG with n non-terminal symbols in it; S corresponds to the PSG's starting symbol; 0 is zero; s(n) is the successor function (i.e. n+1); p is for plus (i.e. p(a,b) is a+b); m is minus (i.e. m(a,b) represents a-b with a lower bound of zero); and J is the string concatenation operator (i.e. J(a,b) is the string of the a as a string followed by b as a string).

$R_0$  also includes rules to reflect the production rules in the PSG.

For each rule in the PSG of the form  $S \rightarrow a_1 a_2 \dots a_n$  there corresponds a SS rule of form

$$\text{PSGi: } S \Rightarrow W(J(a_1, J(a_2, J(\dots J(a_{n-1}, a_n) \dots))), s(s(\dots s(0) \dots))),$$

where  $s(s(\dots s(0) \dots))$  has the same depth as the number of non-terminals in  $a_1 a_2 \dots a_n$ .

For each other rule in the PSG of the form  $a_1 a_2 \dots a_n \rightarrow c_1 c_2 \dots c_p$  there corresponds a SS rule of the form

PSGj:

$$\begin{aligned} & W(J((f, J(J(a_1, J(a_2, J(\dots J(a_{n-1}, a_n) \dots))), g)), s(s(\dots s(0) \dots)))) \\ & \Rightarrow W(J((f, J(J(c_1, J(c_2, J(\dots J(c_{p-1}, c_p) \dots))), g)), p(s(s(\dots s(0) \dots)), i))) \end{aligned}$$

if it increases the number of non-terminal symbols in the PSG, where  $i$  is the increase in the number of non-terminals represented in the form  $s(s(\dots s(0)\dots))$ .

Finally for each other rule in the PSG of the form  $a_1 a_2 \dots a_n \rightarrow c_1 c_2 \dots c_p$  there corresponds a SS rule of the form

PSGk:

$$W(J((f, J(J(a_1, J(a_2, J(\dots J(a_{n-1}, a_n) \dots))), g)), s(s(\dots s(0)\dots)))) \\ \Rightarrow W(J((f, J(J(c_1, J(c_2, J(\dots J(c_{p-1}, c_p) \dots))), g)), m(s(s(\dots s(0)\dots)), i)))$$

if it decreases the number of non-terminal symbols in the PSG, where  $i$  is the decrease in the number of non-terminals represented in the form  $s(s(\dots s(0)\dots))$ .

If one rewrote the above SS rules by:  $w(n)$  for  $W(w,n)$ ; 1 for  $s(0)$ ; 2 for  $s(s(0))$ ; ...;  $n+1$  for  $s(n)$ ;  $b+c$  for  $p(b,c)$ ;  $b-c$  for  $m(b,c)$ ; and  $ab$  for  $J(a,b)$  they would look like:

PSG1:  $\Rightarrow S(0)$

PSG2:  $a(m+(n+1)) \Rightarrow a((m+1)+n)$

PSG3:  $a(b+0) \Rightarrow a(n)$

PSG4:  $a((m+1)-(n+1)) \Rightarrow a(m-n)$

PSG5:  $a(n-0) \Rightarrow a(n)$

PSG6:  $a(bc)(n) \Rightarrow (ab)c(n)$

PSG7:  $(ab)c(n) \Rightarrow a(bc)(n)$

PSGi:  $S \Rightarrow a_1 a_2 \dots a_n(n)$ , where  $n$  is the number of non-terminals,

PSGj:  $a_1 a_2 \dots a_n(n) \rightarrow c_1 c_2 \dots c_p(n+i)$

PSGk:  $a_1 a_2 \dots a_n(n) \rightarrow c_1 c_2 \dots c_p(n-i)$

where  $n$  is the number of non-terminals on the LHS and  $i$  in the increase (respectively decrease) in the number of terminals due to the action of the PSG rule.

Then all the trees of the form  $a(0)$  correspond to the resulting words in the PSG (i.e. those with zero non-terminals in them).

Finally I define a second SS,  $(S_1, R_0)$  with the single rule:

T:  $W(a, 0) \Rightarrow a, (a \in S_0)$

to extract all the trees corresponding to the words generated by the PSG.

### 10.3 Flattening Syntactic Structures

For any finite collection of syntactic structures  $\{S_0, \dots, S_n\}$  which is closed w.r.t. sub-syntaxes, there is a corresponding *flattened* sequence of syntactic structures  $(S_0', \dots, S_k')$ , such that:

1. for each  $S_j$ , there is a  $S_k'$  with a generated set of trees that is identical;
2. for each  $i=1, \dots, k$ , free variables in rules in  $S_i'$  only refer to one other  $S_j'$ , where  $i < j$ .

**Proof Outline:**

Reorder the collection  $\{S_0, \dots, S_n\}$  as  $(T_0, \dots, T_n)$  such that no  $T_i$  has a sub-syntax with a greater index than itself, with the permutation  $\theta: \{1, \dots, n\} \rightarrow \{1, \dots, n\}$ . This is possible since  $\{S_0, \dots, S_n\}$  can be represented as an acyclic digraph.

Ensure that included in the atomic symbols  $A = \{a_1, a_2, \dots, a_n\}$ , there is a distinct symbol for each of  $(T_0, \dots, T_n)$  plus one more, that are not used in any of the rules of any of  $(T_0, \dots, T_n)$ , call these  $t_0, \dots, t_n, n$ .

For each syntactic structure,  $T_i = (SS_i, R_i)$ , in  $(T_0, \dots, T_n)$ , construct another  $T_i' = (SS_i', R_i')$ , to form a new sequence  $(T_0', \dots, T_n')$  such that:

- (a) For each  $T_j$  in  $SS_i$ , put  $T_j'$  in  $SS_i'$ ;
- (b) for each rule  $r_k = (a_1, a_2, \dots, a_m \Rightarrow c, v_1 \in T_{n_1}, \dots, v_p \in T_{n_p})$  in  $R_i$ , put a rule  $r_k' = (t_i(a_1'), t_i(a_2'), \dots, t_i(a_m') \Rightarrow t_i(c'), v_1, \dots, v_p \in T_{i-1})$  in  $R_i'$ , where each  $a_q'$  or  $c'$  is the result of replacing all occurrences of the  $v_1, \dots, v_p$  by  $t_{n_1}(v_1), \dots, t_{n_p}(v_p)$ . Also one extra rule,  $(\Rightarrow v, v \in T_{i-1})$ , to  $R_i'$ .

Essentially the generated results of the syntactic structures are accumulated up the sequence, with the origin of each preserved by the extra symbols.

Finally append a sequence of extra syntactic structures  $(S_1', \dots, S_n')$  which will correspond to the original collection,  $\{S_0, \dots, S_n\}$ , such that  $S_i' = (\{T_{\theta(i)}'\}, \{t_{\theta(i)}(v) \Rightarrow v\})$ . This selects and strips the appropriate elements for the syntax.

## **11 Appendix 4 - A tool for exploring syntactic structures, complexity and simplification**

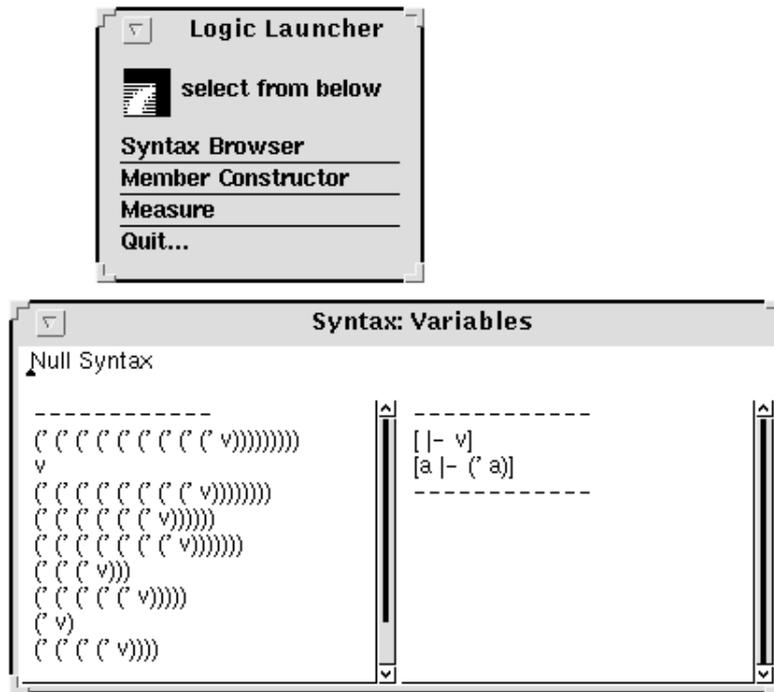
### **11.1 Overview**

In order to aid the exploration of the syntactic structures, I have written a program. This allows the specification of a flattened sequence of interdependent syntactic structures (as specified in section 5.5.4 on page 112 and formalised in section 10 on page 182). Each structure has its own rules, allowing for substitutions from its immediate sub-syntax. Each such structure can then generate its members on demand. Thus this tool implements the automatic generation from an arbitrary dependent chain of syntactic structures as discussed in section 5.5 on page 108 and formalised in Appendix 3 - Formalisation of Syntactic Structure.

### **11.2 Examples**

Here are a couple of simple examples to illustrate its use. Apart from the launching menu each syntactic structure has its own window, with its name in the window title bar, the name of any sub-syntax it might draw substitutions from just below, the list of generating rules on the right and the generated members on the left. Since, in this tool dependency on sub syntaxes is limited to a linear chain, it can be left to the context the symbols in the rules for which substitution is required.

Thus figure 29 shows a simple syntactic structure with two simple rules, to generate an indefinite number of unique items.



**Figure 29.** A window on a single syntactic structure

While figure 30 shows three syntactic levels: the first of which specifies a fixed list of three variable names; the second which generates well formed implicational formulas using the variables syntax and the final level which generates the theories of the implicational fragment of R.

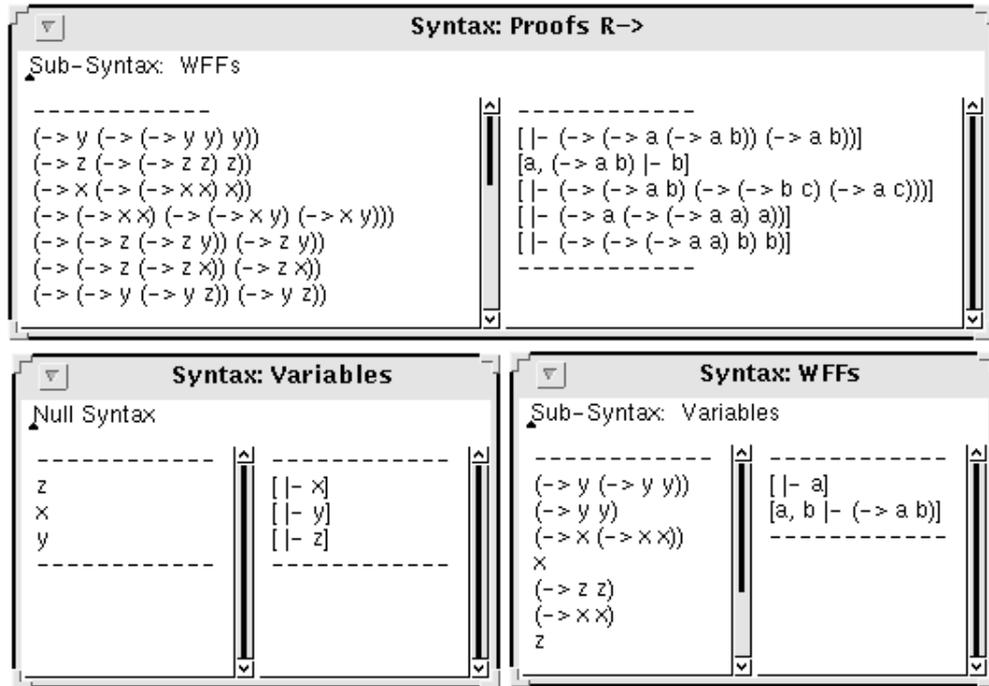


Figure 30. Three inter-dependent syntactic structures in separate windows

### 11.3 Programming

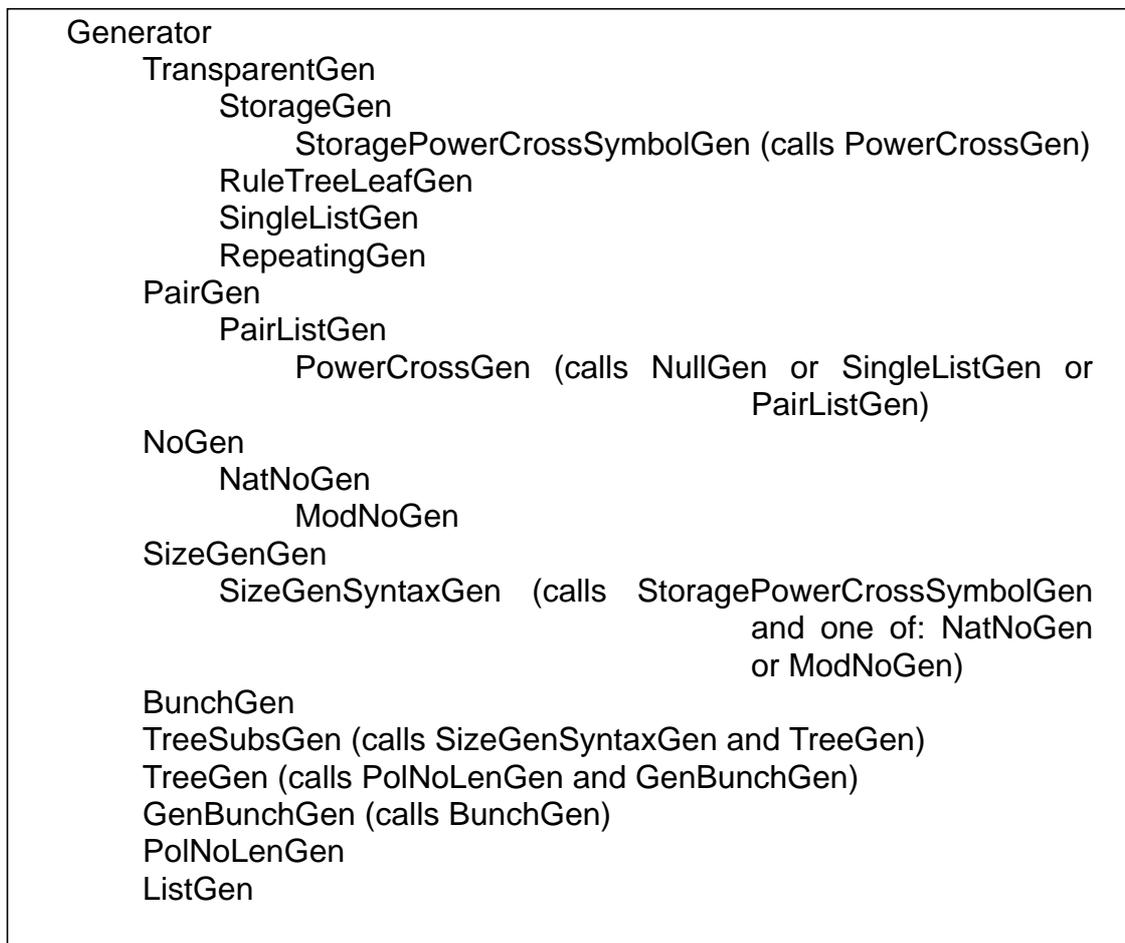
The algorithm was implemented in the object-orientated programming language Objectworks/Smalltalk 4.1. It is composed of a hierarchy of classes to implement the recursive generation of tree structures. Each class in the hierarchy is an incremental generator, in that it only needs to store minimal data in order to produce the next tree in the sequence (in order to avoid an exponential explosion of memory usage). For example, the "TreeGenerator" class uses two other generators: one to generate possible skeletons of valid polish notation strings and the other to generate the possible substitutions of symbols into this skeleton.

Although a few critical generators cache their results for time efficiency (such as the generator that produces the output corresponding to the cross product of two other generators), they act as 'lazy lists', in that elements are only generated as required.

Since Objectworks/Smalltalk 4.1 is an object-orientated language the work in generating from the syntactic structures is done by a series of object instances of classes. Broadly it is the classes that hold the program code and the instances of those classes that

execute the code and store the data. Classes are arranged in a hierarchy so that general code that is placed in a higher class can be ‘inherited’ by lower classes (unless this is overridden). Objects communicate by sending messages to each other.

The generating class hierarchy is show in figure 31 below. Dependency is indicated by the indentation, those other classes it uses are listed after the class name. There are also classes for syntaxes, trees, rules, trees of rules, lists and the empty syntax, but I will not show these are they are either standard or preform simple storage and display functions.



**Figure 31.** The hierarchy of generator classes

Each syntactic structure has generator assigned to it dependent on the type of its rules and whether there is a sub-syntax which it will need to access for substitutions. This could be either a

- RuleTreeLeafGen generator on a ListGen of rules if the rules have no antecedents - this just generates the rules in turn as a rule tree;

- otherwise a `TreeGen` if there is no non-null sub-syntax - this generates all possible legal trees of rules using `PoINoLenGen` to generate polish notation skeletons into which the possible rule substitutions (generated by `SizeGenSyntaxGen`) are slotted;
- and a `TreeSubsGen` if there is - which does the same as `TreeGen` but substutes all combinations of substitutions from lower syntaxes as required.

The full code for all these classes is far to cumbersome to include here but can be read if desired at URL: <http://www.cpm.mmu.ac.uk/~bruce/thesis/code> or supplied on request by mailing me at (b.edmonds@mmu.ac.uk).

## 11.4 Interface

The interface is very rudimentary, consisting only of scrollable passive output in smalltalk windows and input via smalltalk expressions executed on the smalltalk workspace.

## 12 Appendix 5 - A comparison of different rankings of logical formula

size	formula (ignoring formula essentially identical under renaming of variables)
1	$a$
2	$\neg a$
3	$\neg\neg a, a \rightarrow a, a \rightarrow b$
4	$\neg\neg\neg a, a \rightarrow \neg a, \neg a \rightarrow a, \neg(a \rightarrow a), a \rightarrow \neg b, \neg a \rightarrow b, \neg(a \rightarrow b)$
5	$\neg\neg\neg\neg a, \neg(a \rightarrow \neg a), \neg(\neg a \rightarrow a), \neg\neg(a \rightarrow a), \neg a \rightarrow \neg a, a \rightarrow \neg\neg a, \neg\neg a \rightarrow a, (a \rightarrow a) \rightarrow a, a \rightarrow (a \rightarrow a), \neg(a \rightarrow \neg b), \neg(\neg a \rightarrow b), \neg\neg(a \rightarrow b), \neg a \rightarrow \neg b, a \rightarrow \neg\neg b, \neg\neg a \rightarrow b, (a \rightarrow b) \rightarrow a, a \rightarrow (b \rightarrow a), (a \rightarrow b) \rightarrow b, a \rightarrow (b \rightarrow b), (a \rightarrow b) \rightarrow c, a \rightarrow (b \rightarrow c)$
6	$\neg\neg\neg\neg\neg a, \neg\neg(a \rightarrow \neg a), \neg\neg(\neg a \rightarrow a), \neg\neg\neg(a \rightarrow a), \neg(\neg a \rightarrow \neg a), \neg(a \rightarrow \neg\neg a), \neg(\neg\neg a \rightarrow a), \neg((a \rightarrow a) \rightarrow a), \neg(a \rightarrow (a \rightarrow a)), \neg(\neg a \rightarrow \neg a), \neg(\neg\neg a \rightarrow a), \neg\neg a \rightarrow \neg a, \neg a \rightarrow \neg\neg a, \neg\neg\neg a \rightarrow a, a \rightarrow \neg\neg\neg a, (\neg a \rightarrow a) \rightarrow a, \neg a \rightarrow (a \rightarrow a), (a \rightarrow \neg a) \rightarrow a, a \rightarrow (\neg a \rightarrow a), (a \rightarrow a) \rightarrow \neg a, a \rightarrow (a \rightarrow \neg a), \neg(a \rightarrow a) \rightarrow a, a \rightarrow \neg(a \rightarrow a), \neg\neg(a \rightarrow \neg b), \neg\neg(\neg a \rightarrow b), \neg\neg\neg(a \rightarrow b), \neg(\neg a \rightarrow \neg b), \neg(a \rightarrow \neg\neg b), \neg(\neg\neg a \rightarrow b), \neg(\neg a \rightarrow \neg b), \neg(\neg\neg a \rightarrow b), \neg\neg a \rightarrow \neg b, \neg a \rightarrow \neg\neg b, \neg\neg\neg a \rightarrow b, a \rightarrow \neg\neg\neg b, \neg((a \rightarrow b) \rightarrow a), \neg(a \rightarrow (b \rightarrow a)), (\neg a \rightarrow b) \rightarrow a, \neg a \rightarrow (b \rightarrow a), (a \rightarrow \neg b) \rightarrow a, a \rightarrow (\neg b \rightarrow a), (a \rightarrow b) \rightarrow \neg a, a \rightarrow (b \rightarrow \neg a), \neg(a \rightarrow b) \rightarrow a, a \rightarrow \neg(b \rightarrow a), \neg((a \rightarrow a) \rightarrow b), \neg(a \rightarrow (a \rightarrow b)), (\neg a \rightarrow a) \rightarrow b, \neg a \rightarrow (a \rightarrow b), (a \rightarrow \neg a) \rightarrow b, a \rightarrow (\neg a \rightarrow b), (a \rightarrow a) \rightarrow \neg b, a \rightarrow (a \rightarrow \neg b), \neg(a \rightarrow a) \rightarrow b, a \rightarrow \neg(a \rightarrow b), \neg((a \rightarrow b) \rightarrow c), \neg(a \rightarrow (b \rightarrow c)), (\neg a \rightarrow b) \rightarrow c, \neg a \rightarrow (b \rightarrow c), (a \rightarrow \neg b) \rightarrow c, a \rightarrow (\neg b \rightarrow c), (a \rightarrow b) \rightarrow \neg c, a \rightarrow (b \rightarrow \neg c), \neg(a \rightarrow b) \rightarrow c, a \rightarrow \neg(b \rightarrow c)$

**Table 2:** Formulas up to size 6 sorted by size

number of distinct variables	formula (ignoring formula essentially identical under renaming of variables)
1	$a, \neg a, \neg\neg a, a \rightarrow a, \neg\neg\neg a, a \rightarrow \neg a, \neg a \rightarrow a, \neg(a \rightarrow a), \neg\neg\neg\neg a, \neg(a \rightarrow \neg a), \neg(\neg a \rightarrow a), \neg\neg(a \rightarrow a), \neg a \rightarrow \neg a, a \rightarrow \neg\neg a, \neg\neg a \rightarrow a, (a \rightarrow a) \rightarrow a, a \rightarrow (a \rightarrow a), \neg\neg\neg\neg\neg a, \neg\neg(a \rightarrow \neg a), \neg\neg(\neg a \rightarrow a), \neg\neg\neg(a \rightarrow a), \neg(\neg a \rightarrow \neg a), \neg(a \rightarrow \neg\neg a), \neg(\neg\neg a \rightarrow a), \neg((a \rightarrow a) \rightarrow a), \neg(a \rightarrow (a \rightarrow a)), \neg(\neg a \rightarrow \neg a), \neg(\neg\neg a \rightarrow a), \neg\neg a \rightarrow \neg a, \neg a \rightarrow \neg\neg a, \neg\neg\neg a \rightarrow a, a \rightarrow \neg\neg\neg a, (\neg a \rightarrow a) \rightarrow a, \neg a \rightarrow (a \rightarrow a), (a \rightarrow \neg a) \rightarrow a, a \rightarrow (\neg a \rightarrow a), (a \rightarrow a) \rightarrow \neg a, a \rightarrow (a \rightarrow \neg a), \neg(a \rightarrow a) \rightarrow a, a \rightarrow \neg(a \rightarrow a)$
2	$a \rightarrow b, a \rightarrow \neg b, \neg a \rightarrow b, \neg(a \rightarrow b), \neg(a \rightarrow \neg b), \neg(\neg a \rightarrow b), \neg\neg(a \rightarrow b), \neg a \rightarrow \neg b, a \rightarrow \neg\neg b, \neg\neg a \rightarrow b, (a \rightarrow b) \rightarrow a, a \rightarrow (b \rightarrow a), (a \rightarrow b) \rightarrow b, a \rightarrow (b \rightarrow b), \neg\neg(a \rightarrow \neg b), \neg\neg(\neg a \rightarrow b), \neg\neg\neg(a \rightarrow b), \neg(\neg a \rightarrow \neg b), \neg(a \rightarrow \neg\neg b), \neg(\neg\neg a \rightarrow b), \neg(\neg a \rightarrow \neg b), \neg(\neg\neg a \rightarrow b), \neg\neg a \rightarrow \neg b, \neg a \rightarrow \neg\neg b, \neg\neg\neg a \rightarrow b, a \rightarrow \neg\neg\neg b, \neg((a \rightarrow b) \rightarrow a), \neg(a \rightarrow (b \rightarrow a)), (\neg a \rightarrow b) \rightarrow a, \neg a \rightarrow (b \rightarrow a), (a \rightarrow \neg b) \rightarrow a, a \rightarrow (\neg b \rightarrow a), (a \rightarrow b) \rightarrow \neg a, a \rightarrow (b \rightarrow \neg a), \neg(a \rightarrow b) \rightarrow a, a \rightarrow \neg(b \rightarrow a), \neg((a \rightarrow a) \rightarrow b), \neg(a \rightarrow (a \rightarrow b)), (\neg a \rightarrow a) \rightarrow b, \neg a \rightarrow (a \rightarrow b), (a \rightarrow \neg a) \rightarrow b, a \rightarrow (\neg a \rightarrow b), (a \rightarrow a) \rightarrow \neg b, a \rightarrow (a \rightarrow \neg b), \neg(a \rightarrow a) \rightarrow b, a \rightarrow \neg(a \rightarrow b)$
3	$(a \rightarrow b) \rightarrow c, a \rightarrow (b \rightarrow c), \neg((a \rightarrow b) \rightarrow c), \neg(a \rightarrow (b \rightarrow c)), (\neg a \rightarrow b) \rightarrow c, \neg a \rightarrow (b \rightarrow c), (a \rightarrow \neg b) \rightarrow c, a \rightarrow (\neg b \rightarrow c), (a \rightarrow b) \rightarrow \neg c, a \rightarrow (b \rightarrow \neg c), \neg(a \rightarrow b) \rightarrow c, a \rightarrow \neg(b \rightarrow c)$

**Table 3:** Formulas up to size 6 sorted by number of distinct variables

maximum depth	formula (ignoring formula essentially identical under renaming of variables)
0	$a$
1	$\neg a, a \rightarrow a, a \rightarrow b$
2	$\neg\neg a, a \rightarrow \neg a, \neg a \rightarrow a, \neg(a \rightarrow a), a \rightarrow \neg b, \neg a \rightarrow b, \neg(a \rightarrow b), \neg a \rightarrow \neg a, \neg a \rightarrow \neg b, (a \rightarrow a) \rightarrow a, a \rightarrow (a \rightarrow a), (a \rightarrow b) \rightarrow a, a \rightarrow (b \rightarrow a), (a \rightarrow b) \rightarrow b, a \rightarrow (b \rightarrow b), (a \rightarrow b) \rightarrow c, a \rightarrow (b \rightarrow c), (a \rightarrow b) \rightarrow \neg c, \neg a \rightarrow (b \rightarrow c), \neg a \rightarrow (b \rightarrow a), (a \rightarrow b) \rightarrow \neg a, \neg a \rightarrow (a \rightarrow b), (a \rightarrow a) \rightarrow \neg b, \neg a \rightarrow (a \rightarrow a), (a \rightarrow a) \rightarrow \neg a$
3	$\neg\neg\neg a, \neg(a \rightarrow \neg a), \neg(\neg a \rightarrow a), \neg\neg(a \rightarrow a), a \rightarrow \neg\neg a, \neg\neg a \rightarrow a, \neg(a \rightarrow \neg b), \neg(\neg a \rightarrow b), \neg\neg(a \rightarrow b), a \rightarrow \neg\neg b, \neg\neg a \rightarrow b, \neg((a \rightarrow b) \rightarrow a), \neg(a \rightarrow (b \rightarrow a)), (\neg a \rightarrow b) \rightarrow a, (a \rightarrow \neg b) \rightarrow a, a \rightarrow (\neg b \rightarrow a), a \rightarrow (b \rightarrow \neg a), \neg(a \rightarrow b) \rightarrow a, a \rightarrow \neg(b \rightarrow a), \neg((a \rightarrow a) \rightarrow b), \neg(a \rightarrow (a \rightarrow b)), (\neg a \rightarrow a) \rightarrow b, (a \rightarrow \neg a) \rightarrow b, a \rightarrow (\neg a \rightarrow b), a \rightarrow (a \rightarrow \neg b), \neg(a \rightarrow a) \rightarrow b, a \rightarrow \neg(a \rightarrow b), \neg((a \rightarrow b) \rightarrow c), \neg(a \rightarrow (b \rightarrow c)), (\neg a \rightarrow b) \rightarrow c, (a \rightarrow \neg b) \rightarrow c, a \rightarrow (\neg b \rightarrow c), a \rightarrow (b \rightarrow \neg c), \neg(a \rightarrow b) \rightarrow c, a \rightarrow \neg(b \rightarrow c), \neg(\neg a \rightarrow \neg a), \neg((a \rightarrow a) \rightarrow a), \neg(a \rightarrow (a \rightarrow a)), \neg\neg a \rightarrow \neg a, \neg a \rightarrow \neg\neg a, (\neg a \rightarrow a) \rightarrow a, (a \rightarrow \neg a) \rightarrow a, a \rightarrow (\neg a \rightarrow a), a \rightarrow (a \rightarrow \neg a), \neg(a \rightarrow a) \rightarrow a, a \rightarrow \neg(a \rightarrow a), \neg(\neg a \rightarrow \neg b), \neg(\neg a \rightarrow \neg b), \neg\neg a \rightarrow \neg b, \neg a \rightarrow \neg\neg b$
4	$\neg\neg\neg\neg a, \neg\neg(a \rightarrow \neg a), \neg\neg(\neg a \rightarrow a), \neg\neg\neg(a \rightarrow a), \neg(a \rightarrow \neg\neg a), \neg(\neg\neg a \rightarrow a), \neg(\neg\neg a \rightarrow a), \neg\neg\neg a \rightarrow a, a \rightarrow \neg\neg\neg a, \neg\neg(a \rightarrow \neg b), \neg\neg(\neg a \rightarrow b), \neg\neg\neg(a \rightarrow b), \neg(a \rightarrow \neg\neg b), \neg(\neg\neg a \rightarrow b), \neg(\neg\neg a \rightarrow b), \neg\neg\neg a \rightarrow b, a \rightarrow \neg\neg\neg b$
5	$\neg\neg\neg\neg\neg a$

**Table 4:** Formulas up to size 6 sorted by maximum depth

breadth	formula (ignoring formula essentially identical under renaming of variables)
1	$a, \neg a, \neg\neg a, \neg\neg\neg a, \neg\neg\neg\neg a, \neg\neg\neg\neg\neg a$
2	$a \rightarrow a, a \rightarrow b, a \rightarrow \neg a, \neg a \rightarrow a, \neg(a \rightarrow a), a \rightarrow \neg b, \neg a \rightarrow b,$ $\neg(a \rightarrow b), \neg(a \rightarrow \neg a), \neg(\neg a \rightarrow a), \neg\neg(a \rightarrow a), \neg a \rightarrow \neg a,$ $a \rightarrow \neg\neg a, \neg\neg a \rightarrow a, \neg(a \rightarrow \neg b), \neg(\neg a \rightarrow b), \neg\neg(a \rightarrow b),$ $\neg a \rightarrow \neg b, a \rightarrow \neg\neg b, \neg\neg a \rightarrow b, \neg\neg(a \rightarrow \neg a), \neg\neg(\neg a \rightarrow a),$ $\neg\neg\neg(a \rightarrow a), \neg(\neg a \rightarrow \neg a), \neg(a \rightarrow \neg\neg a), \neg(\neg\neg a \rightarrow a),$ $\neg(\neg a \rightarrow \neg a), \neg(\neg\neg a \rightarrow a), \neg\neg a \rightarrow \neg a, \neg a \rightarrow \neg\neg a, \neg\neg\neg a \rightarrow a,$ $a \rightarrow \neg\neg\neg a, \neg\neg(a \rightarrow \neg b), \neg\neg(\neg a \rightarrow b), \neg\neg\neg(a \rightarrow b),$ $\neg(\neg a \rightarrow \neg b), \neg(a \rightarrow \neg\neg b), \neg(\neg\neg a \rightarrow b), \neg(\neg a \rightarrow \neg b),$ $\neg(\neg\neg a \rightarrow b), \neg\neg a \rightarrow \neg b, \neg a \rightarrow \neg\neg b, \neg\neg\neg a \rightarrow b, a \rightarrow \neg\neg\neg b$
3	$(a \rightarrow a) \rightarrow a, a \rightarrow (a \rightarrow a), (a \rightarrow b) \rightarrow a, a \rightarrow (b \rightarrow a),$ $(a \rightarrow b) \rightarrow b, a \rightarrow (b \rightarrow b), (a \rightarrow b) \rightarrow c, a \rightarrow (b \rightarrow c),$ $\neg((a \rightarrow a) \rightarrow a), \neg(a \rightarrow (a \rightarrow a)), (\neg a \rightarrow a) \rightarrow a,$ $\neg a \rightarrow (a \rightarrow a), (a \rightarrow \neg a) \rightarrow a, a \rightarrow (\neg a \rightarrow a), (a \rightarrow a) \rightarrow \neg a,$ $a \rightarrow (a \rightarrow \neg a), \neg(a \rightarrow a) \rightarrow a, a \rightarrow \neg(a \rightarrow a), \neg((a \rightarrow b) \rightarrow a),$ $\neg(a \rightarrow (b \rightarrow a)), (\neg a \rightarrow b) \rightarrow a, \neg a \rightarrow (b \rightarrow a), (a \rightarrow \neg b) \rightarrow a,$ $a \rightarrow (\neg b \rightarrow a), (a \rightarrow b) \rightarrow \neg a, a \rightarrow (b \rightarrow \neg a), \neg(a \rightarrow b) \rightarrow a,$ $a \rightarrow \neg(b \rightarrow a), \neg((a \rightarrow a) \rightarrow b), \neg(a \rightarrow (a \rightarrow b)),$ $(\neg a \rightarrow a) \rightarrow b, \neg a \rightarrow (a \rightarrow b), (a \rightarrow \neg a) \rightarrow b, a \rightarrow (\neg a \rightarrow b),$ $(a \rightarrow a) \rightarrow \neg b, a \rightarrow (a \rightarrow \neg b), \neg(a \rightarrow a) \rightarrow b, a \rightarrow \neg(a \rightarrow b),$ $\neg((a \rightarrow b) \rightarrow c), \neg(a \rightarrow (b \rightarrow c)), (\neg a \rightarrow b) \rightarrow c,$ $\neg a \rightarrow (b \rightarrow c), (a \rightarrow \neg b) \rightarrow c, a \rightarrow (\neg b \rightarrow c), (a \rightarrow b) \rightarrow \neg c,$ $a \rightarrow (b \rightarrow \neg c), \neg(a \rightarrow b) \rightarrow c, a \rightarrow \neg(b \rightarrow c)$

**Table 5:** Formulas up to size 6 sorted by breadth (number of variables)

cyclomatic number	formula (ignoring formula essentially identical under renaming of variables)
0	$a, \neg a, \neg\neg a, a \rightarrow b, \neg\neg\neg a, a \rightarrow \neg b, \neg a \rightarrow b, \neg(a \rightarrow b), \neg\neg\neg\neg a, \neg(a \rightarrow \neg b), \neg(\neg a \rightarrow b), \neg\neg(a \rightarrow b), \neg a \rightarrow \neg b, a \rightarrow \neg\neg b, \neg\neg a \rightarrow b, (a \rightarrow b) \rightarrow c, a \rightarrow (b \rightarrow c), \neg\neg\neg\neg\neg a, \neg\neg(a \rightarrow \neg b), \neg\neg(\neg a \rightarrow b), \neg\neg\neg(a \rightarrow b), \neg(\neg a \rightarrow \neg b), \neg(a \rightarrow \neg\neg b), \neg(\neg\neg a \rightarrow b), \neg(\neg a \rightarrow \neg b), \neg(\neg\neg a \rightarrow b), \neg\neg a \rightarrow \neg b, \neg a \rightarrow \neg\neg b, \neg\neg\neg a \rightarrow b, a \rightarrow \neg\neg\neg b, \neg((a \rightarrow b) \rightarrow c), \neg(a \rightarrow (b \rightarrow c)), (\neg a \rightarrow b) \rightarrow c, \neg a \rightarrow (b \rightarrow c), (a \rightarrow \neg b) \rightarrow c, a \rightarrow (\neg b \rightarrow c), (a \rightarrow b) \rightarrow \neg c, a \rightarrow (b \rightarrow \neg c), \neg(a \rightarrow b) \rightarrow c, a \rightarrow \neg(b \rightarrow c)$
1	$a \rightarrow a, a \rightarrow \neg a, \neg a \rightarrow a, \neg(a \rightarrow a), \neg(a \rightarrow \neg a), \neg(\neg a \rightarrow a), \neg\neg(a \rightarrow a), \neg a \rightarrow \neg a, a \rightarrow \neg\neg a, \neg\neg a \rightarrow a, (a \rightarrow b) \rightarrow a, a \rightarrow (b \rightarrow a), (a \rightarrow b) \rightarrow b, a \rightarrow (b \rightarrow b), \neg\neg(a \rightarrow \neg a), \neg\neg(\neg a \rightarrow a), \neg\neg\neg(a \rightarrow a), \neg(\neg a \rightarrow \neg a), \neg(a \rightarrow \neg\neg a), \neg(\neg\neg a \rightarrow a), \neg(\neg a \rightarrow \neg a), \neg(\neg\neg a \rightarrow a), \neg\neg a \rightarrow \neg a, \neg a \rightarrow \neg\neg a, \neg\neg\neg a \rightarrow a, a \rightarrow \neg\neg\neg a, \neg((a \rightarrow b) \rightarrow a), \neg(a \rightarrow (b \rightarrow a)), (\neg a \rightarrow b) \rightarrow a, \neg a \rightarrow (b \rightarrow a), (a \rightarrow \neg b) \rightarrow a, a \rightarrow (\neg b \rightarrow a), (a \rightarrow b) \rightarrow \neg a, a \rightarrow (b \rightarrow \neg a), \neg(a \rightarrow b) \rightarrow a, a \rightarrow \neg(b \rightarrow a), \neg((a \rightarrow a) \rightarrow b), \neg(a \rightarrow (a \rightarrow b)), (\neg a \rightarrow a) \rightarrow b, \neg a \rightarrow (a \rightarrow b), (a \rightarrow \neg a) \rightarrow b, a \rightarrow (\neg a \rightarrow b), (a \rightarrow a) \rightarrow \neg b, a \rightarrow (a \rightarrow \neg b), \neg(a \rightarrow a) \rightarrow b, a \rightarrow \neg(a \rightarrow b)$
2	$(a \rightarrow a) \rightarrow a, a \rightarrow (a \rightarrow a), \neg((a \rightarrow a) \rightarrow a), \neg(a \rightarrow (a \rightarrow a)), (\neg a \rightarrow a) \rightarrow a, \neg a \rightarrow (a \rightarrow a), (a \rightarrow \neg a) \rightarrow a, a \rightarrow (\neg a \rightarrow a), (a \rightarrow a) \rightarrow \neg a, a \rightarrow (a \rightarrow \neg a), \neg(a \rightarrow a) \rightarrow a, a \rightarrow \neg(a \rightarrow a)$

**Table 6:** Formulas up to size 6 sorted by cyclomatic number of their minimal graph

by number of symbols (size)	by number of different variables	depth	breath	cyclomatic complexity of minimal graph
$a$	$a$	$a$	$a$	$a$
$a \rightarrow a$	$a \rightarrow a$	$a \rightarrow a$	$\neg\neg\neg a$	$a \rightarrow b$
$a \rightarrow b$	$\neg\neg\neg a$	$a \rightarrow b$	$a \rightarrow a$	$\neg\neg\neg a$
$\neg\neg\neg a$	$\neg a \rightarrow \neg a$	$a \rightarrow \neg b$	$a \rightarrow b$	$a \rightarrow \neg b$
$a \rightarrow \neg b$	$a \rightarrow \neg\neg a$	$\neg a \rightarrow \neg a$	$a \rightarrow \neg b$	$a \rightarrow (b \rightarrow (c \rightarrow d))$
$\neg a \rightarrow \neg a$	$a \rightarrow (a \rightarrow a)$	$a \rightarrow (a \rightarrow a)$	$\neg a \rightarrow \neg a$	$a \rightarrow a$
$a \rightarrow \neg\neg a$	$(\neg a \rightarrow a) \rightarrow a$	$a \rightarrow (b \rightarrow a)$	$a \rightarrow \neg\neg a$	$\neg a \rightarrow \neg a$
$a \rightarrow (a \rightarrow a)$	$(a \rightarrow a) \rightarrow (a \rightarrow a)$	$(a \rightarrow b) \rightarrow (a \rightarrow b)$	$a \rightarrow (a \rightarrow a)$	$a \rightarrow \neg\neg a$
$a \rightarrow (b \rightarrow a)$	$a \rightarrow b$	$(a \rightarrow a) \rightarrow (a \rightarrow a)$	$a \rightarrow (b \rightarrow a)$	$a \rightarrow (b \rightarrow a)$
$(\neg a \rightarrow a) \rightarrow a$	$a \rightarrow \neg b$	$\neg\neg\neg a$	$(\neg a \rightarrow a) \rightarrow a$	$(a \rightarrow b) \rightarrow (a \rightarrow b)$
$(a \rightarrow b) \rightarrow (a \rightarrow b)$	$a \rightarrow (b \rightarrow a)$	$a \rightarrow \neg\neg a$	$(a \rightarrow b) \rightarrow (a \rightarrow b)$	$a \rightarrow (a \rightarrow a)$
$(a \rightarrow a) \rightarrow (a \rightarrow a)$	$(a \rightarrow b) \rightarrow (a \rightarrow b)$	$(\neg a \rightarrow a) \rightarrow a$	$(a \rightarrow a) \rightarrow (a \rightarrow a)$	$(\neg a \rightarrow a) \rightarrow a$
$a \rightarrow (b \rightarrow (c \rightarrow d))$	$(a \rightarrow b) \rightarrow (\neg b \rightarrow \neg a)$	$(a \rightarrow b) \rightarrow (\neg b \rightarrow \neg a)$	$a \rightarrow (b \rightarrow (c \rightarrow d))$	$(a \rightarrow a) \rightarrow (a \rightarrow a)$
$(a \rightarrow b) \rightarrow (\neg b \rightarrow \neg a)$	$a \rightarrow (b \rightarrow (c \rightarrow d))$	$a \rightarrow (b \rightarrow (c \rightarrow d))$	$(a \rightarrow b) \rightarrow (\neg b \rightarrow \neg a)$	$(a \rightarrow b) \rightarrow (\neg b \rightarrow \neg a)$

**Table 7:** Rankings of formula by different methods (bold lines group equally ranked formulas)

## 13 Appendix 6 - Complexity and Scientific Modelling

Presented at the 20<sup>th</sup> Wittgenstein Symposium, Kirchberg, Austria, 10-16 August 1997 [150]. References have been combined with main sequence.

### Complexity and Scientific Modelling

by Bruce Edmonds

#### Overview

There have been many attempts at formulating measures of complexity<sup>60</sup> of physical processes (or more usually of the data models composed of sequences of measurements made on them). Here we reject this direct approach and attribute complexity only to *models* of these processes in a given language, to reflect its “difficulty”. This means that it is a highly abstract construct relative to the language of representation and the type of difficulty that concerns one.

A framework for modelling is outlined which includes the language of modelling, the complexity of models in that language, the error in the model's predictions and the specificity of the model (which roughly corresponds to its refutability or to the information it gives about the process).

Many previous formulations of complexity can be seen as either: a special case of this framework using particular modelling languages and focusing on particular types of difficulty; attempts to “objectify” complexity by considering only minimally complex models or its asymptotic behaviour; relativising it to a fixed mathematical structure in the absence of noise.

Such a framework makes sense of a number of aspects of scientific modelling.

*Firstly*, as a result complexity is not situated between order and disorder, as several authors have assumed, but rather such judgements arise given certain natural assumptions about the language of modelling and the desirable trade-offs between the model complexity, its specificity and its error rate.

---

60.Throughout I am distinguishing concerns of complexity from those of *simplicity*, which has come to mean something slightly different in philosophy (see section entitled Complexity and Induction).

*Secondly*, noise can be seen as that which is unpredictable given the available resources of the modeller. In this way noise is distinguished from randomness. Different ways of practically distinguishing noise can thus be seen as resulting from different trade-offs between complexity, error, specificity and the choice of modelling language.

*Thirdly*, Complexity is distinguished from concerns of specificity such as refutability, entropy and information. Complexity is thus seen to have context-dependent relations with such measures but in general is independent from them.

*Lastly*, in this light less complex models are *not* a priori more likely to be accurate, but rather that given the typical structure of expressive modelling languages and our limitations in searching through such languages, choosing the simpler model can be a useful heuristic.

## Complexity

There is an understandable wish to measure the complexity of natural systems rather than just of models of systems, but if natural systems have inherent levels of complexity they are beyond us. In practice there is no upper bound on their complexity, as one can always consider them in more detail or by including more aspects. On the other hand, the effective complexity of systems does depend on our models - the exact motions of the planets may be puzzling when you have to describe them in terms of epi-cycles but appear much simpler in terms of ellipses.

It may be objected that even if such “real complexity” is so intractable, one can still make comparative judgements; i.e. it is natural to judge some natural systems as more complex than others. An example of this is the claim that a cell must be simpler than a whole organism. However in defending such judgements one is forced into relating that which is compared within a common model. It is only once you have abstracted away what you consider to be irrelevant details, that such judgements of relative complexity become evident. Thus such comparative judgements get their force from their grounding in such a descriptive common framework (i.e. model). Otherwise one can object that such a judgement is simply false; e.g. that the chemical interaction of a cell with its environment might well be more complex than the organism’s interaction with its environment. In reply, it may be argued that some such models and frameworks are

privileged but this pre-judges decisions about relevance and so is not helpful to an analysis of complexity and its place in scientific modelling.

In any case complexity is more critically dependent upon the model rather than what is modelled, so we will approach it from this point of view and leave the reader to judge whether this distinction is fruitful in understanding the processes involved.

For our purposes we will define complexity thus: “The difficulty associated with a model's form when given almost complete information about the data it formulates.”. This is a special case of the definition given in [147].

The relevant type of “difficulty” depends somewhat upon your goals in modelling. Different kinds of difficulty will result in different measure of complexity. but here it will indicate the difficulty in finding the model in a search starting at the smallest model forms. This could be size, depth or some other indication of the computation that is necessary to discover it.

## A Framework for Analysing Modelling

To frame the discussion of complexity, I will outline a model of modelling.

Firstly it is important to distinguish between the form and predictive meaning of such models. The models themselves are always held in some form. The set of such possible forms can be considered as a language in its broadest sense - frequently it may correspond closely to an actual language, either natural or formal. I will call this the modelling language.

Such models are amenable to some form of inference, in that they can be used to predict some property given some other information (even if sometimes some of the necessary information is only available after the predicted event). At least some of the information comes from what is modelled in the form of measurements. The models correspond, loosely, to what they model via these predictions.

Thus we distinguish its form from the correspondence between possible information and the predictions that one could infer from it. This set of information along with the respective predictions can be thought of as defining a subspace of the space of all relevant possibilities. I will call this subspace the model's semantics, because one can draw an analogy between a logic's syntax and its semantics in terms of the set of logical models a statement is true for.

The primary way in which these models can be judged is by the degree of correspondence between what is modelled and the predictions of the models - its error. This however does not rule out the default model, that “anything can happen”. Such a model is always trivially correct (and thus is typically chosen as a starting point). Thus we also need an additional goal of preferring the more specific (or refutable) model. I will call this the model's specificity. A modeller with infinite resources and time need only use these two measures as guides in its choice of model. In some cases, of course these dual aims might be in conflict. In a given modelling language one might be forced to choose between a vague but accurate model and a specific but more erroneous one. In some accounts the specificity of models are sometimes left out of analyses of modelling because the types of modelling languages considered are inherently precise.

For us more limited beings, with very distinct practical considerations, the complexity of our models become important [77]. As we shall see below we have to balance the complexity, the error and the specificity of our models. Note that here, complexity is a property of the form of the models while the error and specificity are properties of its corresponding model semantics.

## Other Formulations of Complexity

There is not room to do anything but mention but a few of the other formulations of complexity here. I consider only four here.

The Algorithmic Information of a pattern can be considered as the difficulty of storage when using a Turing machine from encoding it minimally or alternatively the difficulty in finding such a program when working through possible programs in order of length (also see the section entitled Complexity vs. Information). The Turing machine is here used to be as general as possible.

The computational complexity of a problem class is the asymptotic difficulty of computing a solution once a program has been found for a solution (usually either by the time it takes or the memory needed). Thus this a special case where the “difficulty” is relativised to the problem size.

Grassberger [194] defines the Effective Measure Complexity of a pattern as the asymptotic behaviour of the amount of information required to predict the next symbol to the level of granularity. This captures an aspect of the scaling behaviour of the information

required for successful prediction by a markov process model. This thus captures the asymptotic behaviour of a special case of my definition.

The topological complexity described by Crutchfield [122], is a measure of the size of the minimal computational model (typically a finite automaton of some variety) in the minimal formal language in which it has a finite model. Thus the complexity of the model is both ‘objectivised’ by considering only minimal models but also related to the fixed hierarchy of formal languages. Crutchfield also defines a measure of specificity similar to Grassberger’s measure, as complementary to the topological complexity.

In each case the desire to attribute complexity purely objectively to a physical process seems to force a relativisation to either some framework for which privilege is claimed (e.g. a Turing Machine), to some aspect of the problem (e.g. granularity of representation) or by considering only the minimal size. This, of course, does not completely eliminate the inherent subjective effects in the process of modelling (principally the language of modelling), and obscures the interplay of complexity, specificity and the error involved.

## Order and Disorder

It has been frequently asserted that complexity lies somewhere between order and disorder [194] (what is sometimes called “the edge of chaos” [249]). Thus in figure 32 below many people (especially it seems physicists<sup>61</sup>) judge that the middle pattern is more complex than the other two.

---

61.The accompanying text from [194] is revealing: “*Some people hesitate between the middle and right panels when being asked to point out the most complex one. But once told that the right one is created by means of a (pseudo-) random number generator, the right panel is usually no longer considered as complex.*” (page 491).



**Figure 32.** Three patterns - after [194]

This idea seems to have dominated the literature about complexity in physics. However, as I shall argue, for such a situation to be true you at least need some other assumptions. As can be seen, in the modelling framework I have described there is absolutely no need for this to be true. Although the highly ordered data might well correspond to the simplest models, it will also often be the case that the most disordered data corresponds to the most complex model forms.

To see this possibility consider the following situation. A modeller has an infinite and precise symbolic language with a limited number of symbols and some fixed grammar such that it includes some small expressions, but expressions of increasing size can be constructed. Suppose this language describes members of a class of data strings of any length of any sequence of symbols taken from a fixed alphabet.

A counting argument shows that most such patterns are disordered (as defined by something like Shannon information or algorithmic information measures), but a similar argument shows that only a few of these patterns can correspond to models with relatively small minimal representations. That is, most of the disordered patterns will correspond to models with relatively large minimal representations. Whatever the ordering in terms of ease of search, in general the bigger forms will be more difficult to find, i.e. more complex.

Thus, in this case, far from complexity and disorder being antithetical, one would be hard pushed to arrange things so that a significant proportion of the most complex models would correspond to even slightly ordered patterns.

So if complexity does not necessarily lie between order and disorder, where has our intuition gone wrong? Without any prior knowledge about the process that produces the

data we have no reliable way of distinguishing what is merely very complex behaviour and what is irrelevant noise. The diagrams above mislead us because our experience about the patterns we typically encounter, has led us to think we recognise the noise, and separate it out from the relevant pattern. That this is not necessarily so, see figure 33, where we suggest that each pattern as a magnification of a section of the one to its right.



**Figure 33.** The same patterns with some suggested inclusions

Faced with this new information one might change one's mind and say the rightmost pattern is the most complex. The initial judgement of the middle pattern comes not because such disordered patterns do not correspond to complex models in some precise languages, but because we are beings with limited resources used to receiving noisy data. We know it is not usually sensible to try to describe such patterns in such precise languages. This is for two reasons: *firstly* we do not have the time and *secondly* broader experience has taught us such models do not predict well on new data. In other words, we know that such “overfitting” of data is not likely to be a profitable activity. The association of what is apparently disordered with simplicity is thus the result of applying a natural heuristic and thus does not represent a necessary relationship.

Introducing specificity into the account makes sense of this. We are naturally good at distinguishing noise from relevant information, so that we do not realise when we are doing it. The moral is that it is extremely useful to use less precise languages to describe such patterns, even at the cost of some accuracy with respect to the data. If the language allows expressions of varying degree of precision, then an overfitting model may well be more complex than a more suitable and less accurate one. The most appropriate model for a pattern with high entropy might be a very simple and very vague model. If the complexity of the patterns is judged by the complexity of their most suitable model one

comes to the intuitive judgement that complexity is “between” order and disorder. It is only when one takes the unnatural step of leaving the specificity out of the account (e.g. by restricting the models to a uniformly precise language) that one is faced with the unnaturalness of the situation where the “best” model for a disordered pattern has to be a very complex one, since it has to be precise.

## Noise

The above account illustrates the importance of making a judgement as to what in the data may be considered as noise. If we take noise as what is effectively unpredictable in a pattern, then we will see that different approaches and conceptions of noise naturally emerge given different responses to excess error.

Let us imagine we are faced with an unacceptable level of error in the predictions of our best current model. What could be done?

*Firstly*, we could search for more accurate models by widening the search to look at other equally precise models. It is sensible to try the easier models first, but if we exhaust all the models at our current level of complexity we will be forced to try more complex ones. In this case we are effectively discounting the case that the unexplained elements of the data are unpredictable and treating noise as what is merely *currently* unexplained due to its complexity. This is a view taken in the light of many chaotic processes which can produce data indistinguishable from purely random data.

*Secondly*, we could decide to look for models that were less specific. This might allow us to find a model that was not significantly more complex but had a lower level of predictive error. Here we are essentially filtering out some of the data, attributing it to some irrelevant source. This might correspond to a situation where you know that there is an essentially random source of noise that has been imposed upon the data. This is the traditional approach, used in a wide range of fields from electronics to economics.

*Thirdly*, and most radically, we could seek to change our language of modelling to one that we felt was more appropriate to the data. Here we have noise as the literally indescribable. For example, sometimes a neural network is set up so that extreme fluctuations in the data are not exactly capturable by the range of functions the network can output. In this way it is forced to approximate the training data and overfitting is avoided.

Thus randomness may be a sufficient characterisation of noise but it is not a necessary one.

## Complexity vs. Information

The above framework distinguishes between the complexity of the model form and its specificity. The specificity of a model has been characterised in many ways, including: the information a model provides, the system's entropy, and the model's refutability.

Such measures of specificity have often been linked to a model's *simplicity*, where by *simplicity* we mean *that property of a model which makes it more likely to be true than another, given that they have equal evidential support*. This property is called "simplicity", because it is traced back to the principle of parsimony attributed to William of Occam. Thus Popper characterises simplicity as a model's refutability [358], while Sober has associated it with the minimum extra information to answer a given question [418]. This tradition has been continued by several authors who have used various measures of information to capture it including Shannon information and algorithmic information<sup>62</sup>. It is clear that such *simplicity* is not necessarily the opposite of complexity, as described above (see section entitled Complexity and Induction).

That complexity is not rigidly linked to the specificity of a model can be shown by considering any modelling language which has terms explicitly denoting non-specificity (frequently called "error terms"). Clearly, the introduction of such terms can make an expression simultaneously more complex and less specific.

This is not to say that there might not be good reasons to prefer a model which is more specific, just that it is neither directly linked to either a model's complexity or its error rate. Rissanen makes a case for a particular trade-off between the specificity of a model and its complexity - namely that one should seek the size of the minimal description which includes the model and the deviations from the model.

## Complexity and Induction

From this framework it is clear that a lack of complexity is, in general, not a reliable guide to a model's error rate<sup>63</sup>, unless the problem and modelling language happen to be

---

62. Variously attributed to combinations of Solomonoff [419], Kolmogorov [266] and Chaitin [99].

63. In this I concur with [364] and [349].

constructed that way<sup>64</sup>. On the other hand experience has frequently shown us that the less complex theory often turns out to be more useful. The answer to this riddle becomes clear when one takes into account the process whereby models are typically developed.

An ideal modeller without significant resource restrictions might well be able to attempt a fairly global search through possible model forms. Some automatic machine-based systems approximate this situation and there it does indeed seem to be the case that smaller complexity is not a reliable guide to truth (e.g. [336]).

Usually, however, and certainly in the case of human modellers they do not have this luxury. They can check only a very limited number of the possible model forms. Fortunately, it is frequently the case that the meaning of the models allows us to intelligently develop and combine the models we have and so produce new models that are much more likely to produce something useful than a typical automatic procedure. Thus it is frequently the case that it is sensible to try elaborations of known models first before launching off into unknown territory where success is, at best, extremely uncertain.

On its own elaboration is, of course, an inadequate strategy. One can get into a position of diminishing returns where each elaboration brings decreasing improvements in the error rate, but at increasing cost. At some stage preferring simpler and more radically different models will be more effective. Thus sometimes choosing the simpler model, even if less precise and accurate is a sensible heuristic, but this is only so given our knowledge of the process of theory elaboration that frequently occurs.

## Conclusion

Complexity is usefully distinguished from both the probability of correctness (the amount of error in its predictions) and the specificity of the model. It is relative to both the type of difficulty one is concerned with and the language of modelling. Complexity does not necessarily correspond to a lack of “*simplicity*” or lie between order and disorder.

When modelling is done by agents with severe resource limitations, the acceptable trade-offs between complexity, error and specificity can determine the effective relations

---

64. For example, if we knew that nature had developed a certain class of systems starting simply and elaborating from this and our language of modelling reflected this structure we would expect there to be a threshold of complexity such that models below this were more likely. Such a principle would then be a contingently true.

between these. The characterisation of noise will emerge from this. Simpler theories are not a priori more likely to be correct but sometimes if one knows that the theories are made by an agent, for whom it is easier to elaborate than engage in a wider search, preferring the simpler theory at the expense of accuracy can be a useful heuristic.

## 14 Appendix 7 - Complexity and Economics

A paper presented at the Workshop on Economics and the Sciences of Complexity, Namur, Belgium, 1997 [149]. References have been combined with main sequence.

### From Complexity to Agent Modelling and Back Again - some implications for economics

Bruce Edmonds

#### What is Complexity?

##### *“Complexity” in economics*

Since Herbert Simon’s distinction between substantive and procedurally rationality [415], complexity in economics has (until recently) simply meant *not* assuming that an economic agent acted as if it had the computational resources to completely cope with the demand placed on it by its environment (for a survey of this area see [229]). In a sense this is intended as a negative definition to contrast with the usual simplifying assumptions of mainstream economics. The use of the word “complexity” here reflects the greater analytic difficulty which is caused by relaxing these assumptions. However, as generally used, this usage is vague - it does not necessarily distinguish between different kinds of limitations on the agent (memory, inference, ability to generalise, access to information, etc.) or different kinds of overloading caused by the environment. It also encourages the conflation of the complexity from the agent’s point of view and from the modeller’s view. Thus although it has performed a useful role in critique it is not a very helpful usage for more focused analytic discourse.

##### *The “Sciences of Complexity”*

Recently there has been an explosion of fields and techniques loosely grouped around the “Sciences of Complexity” banner. Even though these have been considerably over-hyped they include many new useful techniques and metaphors. They are not, however, based on any coherent “complexity theory”. Key practitioners in these fields recognise this and have frequently exhorted, anticipated and worked towards such a theory (e.g. [89, 249]), but at the moment there is no such body of general theory which has a useful analytical meaning outside individual component fields.

What these widely differing areas *do* share is a tendency to:

- use newer formal techniques (logics, automata, topological models, etc.);
- deal with systems where the behaviour of interest arises, at least partly, out of some contingent occurrences (e.g. evolutionary computation);
- tend to use modelling techniques which tend to predict and capture mainly second-order properties (e.g. it may not be able to predict a value at a particular future time but may be able to tell you some statistical properties or reveal some qualitative aspects of the process).

Thus in a broad sense these techniques try to capture some more abstract properties of systems in areas where the application of more traditional techniques is not feasible. Thus this is also a negative definition, indicating that newer approaches had had to be applied. Again, if one needs to attempt a more exact and meaningful analysis this is an inadequate characterisation of the term.

#### *Complexity per se*

So what is complexity *per se*? Let us approach this via a series of considerations.

*Firstly* one has to distinguish what complexity *is* and what may *cause* it. Without an idea of the former it will be very difficult to get a clear idea of the later. In general there will be *many* possible cause of complexity and there well may be no *overall* characterisation of such causes.

*Secondly*, I argue that complexity is not a property *usefully* attributed to natural systems but only to our models of such systems. The reasons for this include:

- If natural systems have a complexity it would be beyond us; it is always possible to arbitrarily increase the complexity of a system by considering more aspects or more detail;
- The complexity of things varies *critically* with the model chosen;

In this way it is similar to the property of “primality” - primality is a property of (some) numbers and not of things enumerated by numbers, even though the selection of groups to be represented by numbers can effect whether the property holds (merely because it can change the number being considered).

*Thirdly*, the complexity is a comparative thing, frequently we want to be able to say “A is more complex than B which is, in turn, more complex than C”.

*Fourthly*, complexity is relative to the framework you are modelling in. This includes the language of representation of your model, your general framework (what is given what you are trying to formulate) and your goals in modelling.

*Finally* complexity is usefully distinguished from ignorance. Ignorance can cause complexity in many ways, including the mis-framing of a problem, and complexity can certainly cause ignorance, for example where there are only limited problem solving resources available. However, it is easy to give examples where the two concepts diverge.

Packaging these all up into a definition we get:

*“Complexity is that property of models which make it difficult to formulate its overall behaviour in a given language of representation, even when given almost complete information about its components and their inter-relations” [147]*

Thus I have relativised complexity to the model, the type of difficulty, the language of modelling and how one characterises “overall behaviour” and “atomic components”. Many of the confusions that have abounded with the use of this word have occurred because of unshared assumptions about these relativisations. Thus you will get different kinds of complexity for different kinds of “difficulty”, different modelling languages etc.

Frequently the complexity of a pattern, system or data model is taken to be the complexity of the most “suitable” model given a certain framework. The definition above does not mean that you can make complexity to mean whatever you want. When a framework has been agreed (either explicitly or, as frequently occurs in the hard sciences, implicitly), complexity can be objectively measured and attributed, just as whether the number of cows in a particular field is prime is an objective question despite the fact that “primality” refers to our (numerical) model of the cows and not the cows themselves.

Such a definition imposes some obligation to explicitly make clear the chosen relativisations but this is no bad thing. The justification of such an approach to defining complexity ultimately comes down to the extent it makes the analysis of the effects and causes of complexity clear (for more on this see [150]). We will now apply it to modelling agents which economically interact.

## The effects of complexity on modelling by agents

We will define “environmental complexity” as a special case of the above definition as:

*The difficulty of making correct predictions about its environment (measured by its error rate) for an agent using the best model it can infer from the information available to it given its computational resources.*

This is a more precise version of the “economic complexity” described in Section . We will define the “model complexity”, pertaining to a particular model of an agent as:

*The computational difficulty of reaching and testing a model given the constraints of the language and the known data it has to fit.*

I will consider four situations representing cases of increasing environmental complexity. The environmental complexity will affect how the agent needs to evaluate its models, in particular the kind of trade-offs between the model complexity, specificity and error.

### *Ideal rationality and perfect information*

If an agent is in a situation where it effectively has all the time it needs for relevant computation and learning with respect to its environment (e.g. if the environment is relatively static and the agent has a long time between decisions or if the whole population acts as a sort of massively parallel search and inference mechanism and the agent has access to the “results”) then it can be treated as if it had ideal rationality and perfect information. If you have an agent in such a position then the only relevant criteria there is for judging alternative models about its environment is that of the *accuracy* (or conversely error) of their predictions.

### *Ideal rationality and noisy information*

A slight increase in the environmental complexity for the agent it when is has noisy data but enough of it to determine the extent of this noise. This requires only a slight increase in sophistication by the agent - it has enough data and computational resources to determine the form of the correct model but might need to estimate its parameters. This situation has been studied in economics as representing a simple learning process. Here

the criterion of accuracy is not needed only for the determination of its best model but also as a characterisation of its resulting behaviour.

*Ideal rationality and inadequate information*

Let us increase the prediction complexity further by now only giving it inadequate (i.e. noisy and insufficient) information about its environment. Let us assume that the agent can posit deliberately imprecise models, i.e. it can include room for inexact predictions using some mechanism like error terms in its language of representation<sup>65</sup>. Now the most appropriate model an agent can infer will typically be, at least somewhat, imprecise, so that as well as *accuracy* the agent also has to take into account the *specificity* of its models. Since the agent has insufficient information it has no way of certainly distinguishing noisy data from very complex behaviour so there will be an inevitable trade-off between the accuracy and specificity of candidate models. Although some [378] have argued that particular trade-offs can be principled, in general the nature of this trade-off will depend on the goals of the agent (e.g. its tolerance to risk).

*Bounded rationality and inadequate information*

Finally we come to the situation which is most environmentally complex. Here, not only does the agent have inaccurate and insufficient information but also that it does not have the computational resources to search the space of possible models for the optimal one. In other words the *model complexity* has to be taken into account along with *accuracy* and *specificity* in the evaluation of candidate models, for there will be a limit to the complexity of models it can consider. Now we have a three-way trade off to consider. - the nature of this will depend upon the agent's goals (some examples of this are given in section entitled The complexity, specificity, error trade-off).

---

65.If it only had a completely precise modelling language (i.e. its models could only make exact predictions) it would tend to overfit noisy data, unless the language was deliberately restricted to avoid this.

## The effects of modelling by agents on complexity

For us modellers of these agents, there is the difficulty of capturing an agent's (or population of agent's) resulting behaviour. I will call this the "behavioural complexity", which I define as:

*The difficulty of finding a model of the overall population's agents behaviour given knowledge of the setup, algorithms and structure of each agent.*

The need to include more about the process of modelling by agents in our models of them increases the behavioural complexity of such agents.

### *Ignoring the process of modelling by economic agents*

Since *we* are agents with distinctly bounded rationality with access only to imperfect information (as described in the section entitled Bounded rationality and inadequate information), it makes sense for us to start with simple models and only progress to more complex ones if we need to, i.e. if our simpler models have inadequate predictive and explanatory power. If the agents are in a situation where the environment is changing sufficiently slowly and the agent has effective access to good information one can conflate the agent's best model of its environment and the true model of the environment, as the agent has plenty of time and resource to effectively estimate the true model.

In such circumstances it is sensible for us to choose the simpler model and ignore the agent's modelling process - it is notable that if *we* did not have distinctly bounded rationality and inadequate information there would be no reason for us *not* to search through models of all complexities for the most accurate one.

### *Including the process of modelling by economic agents*

The other case is where the conditions described above do not hold, i.e. where the process whereby the agent models its environment itself has significant effects. That such cases may exist in real life is surely not in doubt (e.g. in emerging markets, during periods

of great volatility in the stock market or the behaviour of economists themselves). What is debateable is:

1. *the extent of these situations;*
2. *the tractability of them;*
3. *whether such an area of study can be said to be part of the tradition of economics as it has developed.*

Although point (1) is the most important, to be decided by empirical evidence, I do not have either the knowledge or the time to consider this. I will only comment that since the assumption that one does not lose anything important by conflating the agents model with the true model is a very strong one, the burden of evidence should weigh more heavily on it as opposed to its converse.

Point (2) is a valid point, and one which I am highlighting here in this paper. The introduction of including processes of modelling by agents in our models, *greatly* increases the behavioural complexity of the situation. In fact such an introduction seems to introduce a qualitative jump in the whole enterprise of such modelling<sup>66</sup>. On the other hand it does seem, that at least some aspects of such situations are meaningfully formalisable and analysable (for example both [22] and [134] exhibit credible models of economic systems which exhibit both rational expectations and more complex dynamics in clearly defined circumstances).

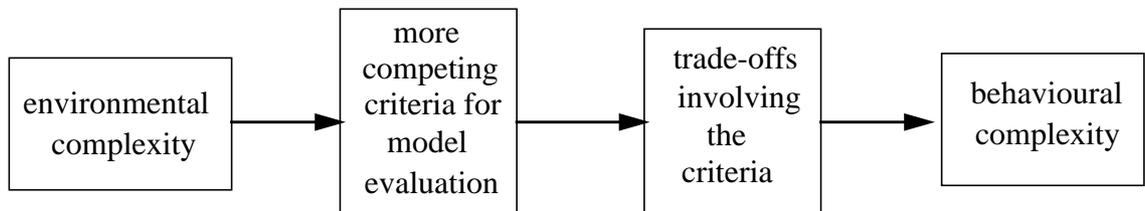
Point (3) is in many ways trivial. The qualitative leap highlighted above may justify categorising such modelling under a separate heading, but unless such models turn out to be significantly less successful than more traditional ones, this would not justify this not being a substantial and legitimate line of research.

---

66.Comparable, perhaps to the qualitative difference induced by the presence of some self-modelling ability.

## Towards dealing with the complexity of modelling agents - modelling modelling

Thus I have built up a picture of the connections in this context between types of complexity:.



**Figure 34.** A diagram of the relationship between relevant types of complexity

If what I have said is correct, we are faced with a task of great behavioural complexity. So how can we proceed? In particular how can we succeed in a way that is not entirely specific to each situation? Such an enterprise comes down to nothing less than establishing a framework for modelling the process of modelling itself. Below I briefly discuss some of the key features that such a framework might have (for more details on this see [333]).

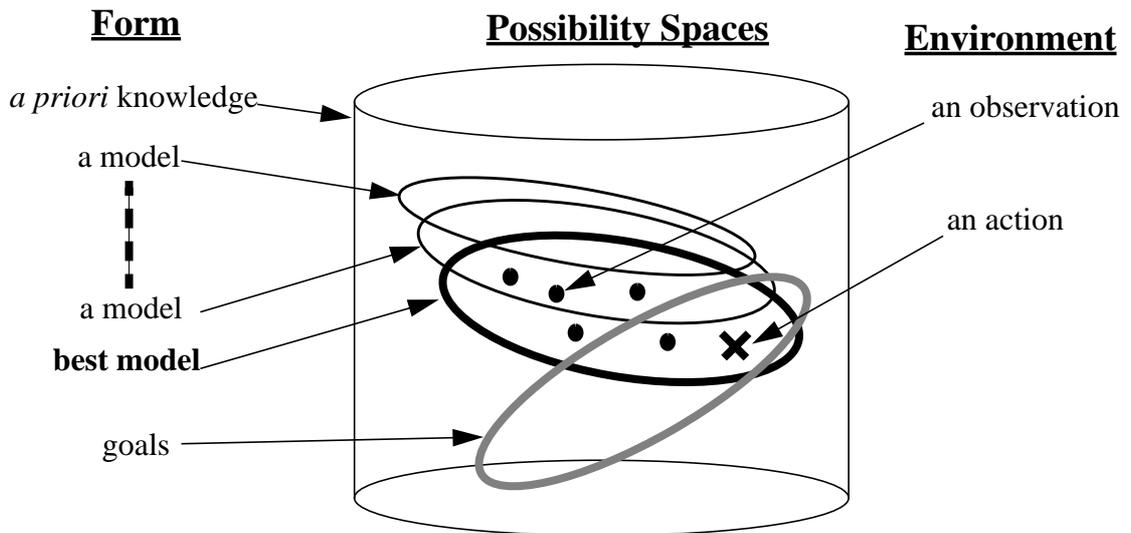
### *The Form - meaning distinction*

The most fundamental distinction, as discussed above, is between the form of an agent's model and its meaning. Where the *meaning* of a model is determined by the mapping of the input and resulting predictions to the model. This is very similar to the syntax-semantics distinction in formal logic<sup>67</sup>. This distinction is critical where we are dealing with agents of bounded rationality. For example, although two models may be equivalent in terms of its predictions, one could be so costly to use that it may be all but useless in deciding on an appropriate action.

In a very general way this can be formalised by reference to the space of all relevant possibilities (similar to phase space diagrams used in physics). Models, goals, a priori knowledge, observations, goals and actions can all be associated with subspaces of this (see figure 35).

---

67. In fact this distinction can be used to induce a logical structure between the agent's models [333].



**Figure 35.** Form and meaning distinction

*The complexity, specificity, error trade-off*

As mentioned above, in situations of greater environmental complexity it is sensible for the agent to accept some trade-offs between model complexity, specificity and error.

There are several desirable formal restrictions on these measures (which can be found in [333]) and there are philosophical justifications and consequences for this three-dimensional analysis of models [150]. Here I will just look at a three examples to illustrate the different possibilities that can result.

1. *A risk-averse agent might take the error rate as the primary criterion whilst accepting the model complexity as representing only a limitation on its resources. It would be very tolerant to vagueness (i.e. low specificity) only accepting a more specific model where one could be found without losing any significant accuracy. This would be particularly appropriate in safety-critical situations.*
2. *On the other hand if an agent is merely trying to predict some out-of-sample data as well as possible on average, then there is no particular reason not to choose quite a specific model. In addition (and depending somewhat on the problem domain) one*

*might not want to choose an overly complex or simple model for fear of overfitting or overpredicting (e.g. as illustrated by [336]).*

3. *Finally, if an agent had an external source of candidate models and it knew that there would have been a tendency to elaborate (make more complex) these when they were relatively unsuccessful on in-sample data, applying a heuristic of preferring simplicity when the evidence is equal would be appropriate.*

#### *The modelling language*

The collection of all possible model forms can be considered as a *language* in its broadest sense. This might correspond to a natural or formal language, but might also be something like a range of possible real-valued vectors (as in some connectionist models). What is clear is that the modelling language can critically effect the nature and “success” of the modelling process. For example if the agent has a relatively inexpressive language that can only capture some of the real possibilities then it can be genuinely surprised by observations from the environment (as distinct from merely experiencing what it considered to be a low probability event). Here it literally can not model certain combinations of possibilities, even if it *can* model the whole possibility space.

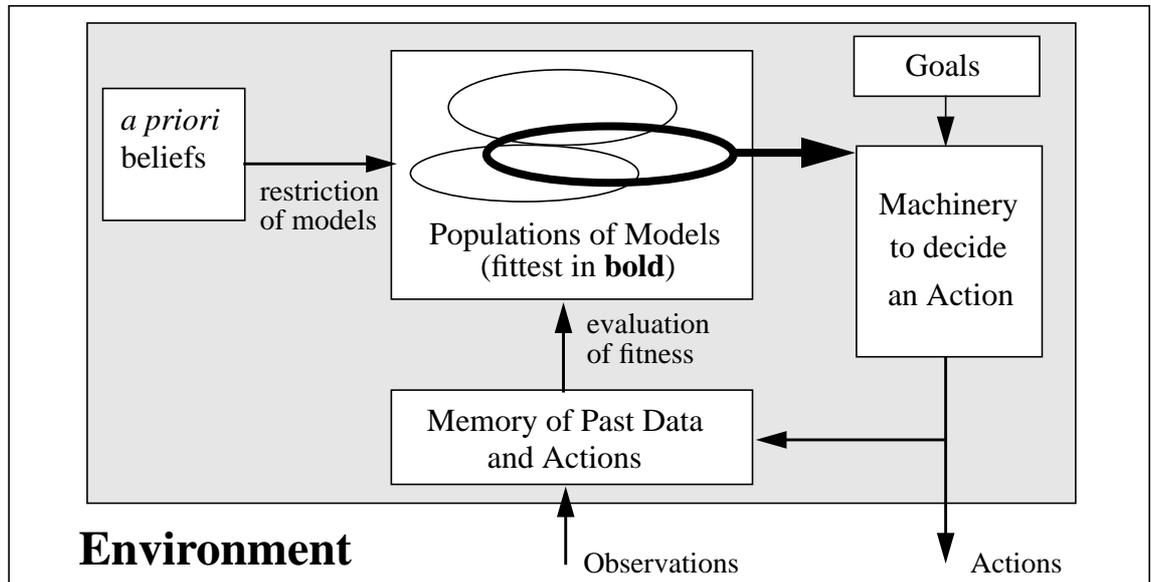
For example, in chaotic processes one can be forced to models of ever greater size as one approaches the critical point at onset of chaos if one restricts oneself to using finite automata but this is simply handleable when using [122]. Despite its importance in learning processes and calls for it to be further studied [393], very little is known about the effect of different modelling languages.

#### *Processes of model development*

So far I have talked about the universe of such models (the modelling language) and the way candidate models are evaluated (the complexity-specificity-accuracy trade-off). The last main aspect of capturing such modelling is the mechanism by which new models are developed.

Despite the fact that there are a multitude of candidate mechanisms to choose from in AI and cognitive science two mechanisms have dominated economic models of learning: those of optimization and evolution (GA/GP/EP etc.). Of these approaches those based on the GP paradigm [270] appear the most appropriate, due to the potentially expressive format of the genome. Using such an approach one can represent an agent as a

population of competing mental models [148]; a possible architecture for such an agent is shown in figure 36.



**Figure 36.** Using a genetic population to model an economic agent

The evolutionary approach has the weakness that the development of variation is *blind* to the process of selection. In other words it avoids the intentionality inherent in human learning. There are, however many other credible models for learning (e.g. [108]).

### Some future directions for economic modelling

I will end with some speculation as to possible future avenues that the modelling of behaviourally complex systems could go.

#### *Applying our model of modelling to ourselves*

Finally let us apply the above model of modelling to ourselves as modellers with bounded rationality and inadequate information. Further let us suppose that our current economic models gave us an unacceptable level of error when applied to situations of

greater environmental and hence behavioural complexity. What avenues might this suggest?

1. *Accept the level of error as inevitable.*
2. *Trade in specificity for accuracy, i.e. accept vaguer models of the same type to decrease error.*
3. *Trade simplicity for accuracy, i.e. expend more effort in search for more complex models of the same type.*
4. *Change the modelling language, i.e. consider new types of models.*

These are roughly ordered in terms of increasing optimism. (1) represents a position of extreme pessimism, and is probably only held by a few holists. (2) is a classical defensive posture for the less specific your models are the less *refutable* they are. (3) represents an optimistic approach - it assumes that the type of model is fine but just needs refining (e.g. by adding a “missing” variable). (4) represents the most radical response, appropriate when the existing modelling language can not even describe the behaviour to be modelled. It is perhaps this option that is usually what is indicated by authors when the “complexity” banner is unfurled.

*Relatively new (non-numerical) techniques*

The history of economics has been dominated by the use of numerical, statistical and game-theoretic modelling languages. Here are three other possibilities that have been taken up by the new sciences:

1. *Formal logics, which allow for a much greater expressivity, especially when one is dealing with qualitative as well as quantitative properties. For example in [311] the assumptions behind some organisational theories are examined. Although these are difficult to use, computer based tools to aid in their use are becoming increasingly accessible for the normal user (e.g. SDML [151]).*
2. *Network based models can be used to capture topological properties in a flexible way, for example in [334] we produced a network model of R&D development which*

*captured some of the context-sensitive nature of the dependencies between technologies.*

- 3. Abstract formal languages, can used to capture generative and modelling behaviours using artificial grammars. I am not aware of an application in economics, but this has been applied to chaotic process [122] and in biology [196].*

### Conclusion - complexity again

I have analysed complexity and distinguished three different kinds of complexity relevant to the modelling of economic agents: environmental complexity, the complexity of an agent's models and the behavioural complexity. The move to considering situations of greater environmental complexity leads to the consideration of agents that evaluate their models using more than just the accuracy of their models and this leads to a greater behavioural complexity. In response to such complexity we may need to use new styles of models that can capture the new behaviour and processes, i.e. the modelling language we use might need to become more sophisticated and hence also the models we build in it.

We have gone from a more closely defined version of economic complexity - environmental complexity all the way to the "sciences of complexity". All these kinds of complexity are *linked* but are far from identical. Hopefully this paper has elucidated the nature of these links.

## 15 References

Many of these references are available electronically in the hypertext bibliography of complexity measures I maintain at URL <http://www.cpm.mmu.ac.uk/~bruce/combib>. This is more convenient as it includes extensive cross-referencing, indexing, a search facility and comments.

- [1] AADERAA, SO; BÖRGER, E. The Horn Complexity of Boolean functions and Cook's problem. In MAYOH, B; JENSEN, F (Eds). *Proceedings of the 5th Scandinavian Logic Symposium*. Aalborg: Aalborg University Press, 1979, 231-256.
- [2] AADERAA, SO; BRÖGER, E. The equivalence of Horn and network complexity of Boolean functions. *Acta Informatica*, 1981, 15, 303-307.
- [3] ABRAHAM, NB; ALBANO, AM; TUFILLARO, NB. Complexity and Chaos. In ABRAHAM, NB (Ed). *Measures of Complexity and Chaos*. New Jersey: Plenum Press, 1989, 1-25.
- [4] ACKERMANN, R. Inductive Simplicity. *Philosophy of Science*, 1961, 2, 152-160.
- [5] ACHINSTEIN, P. *Concepts of Science: A Philosophical Analysis*. Baltimore: Johns Hopkins Press, 1968.
- [6] ACHINSTEIN, P. History and Philosophy of Science: a reply to Cohen, P. In SUPPE, F (ed.). *The Structure of Scientific Theories*. Urbana, IL: University of Urbana Press, 1973, 350-360.
- [7] ADAMI, C; CERF, NJ. Complexity, Computation and Measurement. Fourth Workshop on Physics and Computation, Boston University, 1996. In TOFFOLI, T; BIAFORE, M; LEAO, J. *PhysComp96*, NE: New England Complex Systems Institute, 1996, 7-11.
- [8] AIDA, S et al. *The Science and Praxis of Complexity*. Tokyo: United Nations University, 1985.
- [9] ALBIN, PS. The Complexity of Social Groups and Social Systems by Graph Structures. *Mathematical Social Sciences*, 1980, 1, 101-129.
- [10] ALBIN, PS; GOTTINGER, HW. Structure and Complexity in Economic and Social Systems. *Mathematical Social Sciences*, 1983, 5, 253-268.
- [11] ALLEN, GM; GOULD, EM. Complexity, Wickedness and Public Forests. *Journal of Forestry*, 1986, 84, 20-23.
- [12] ANDERSON, AR; BELNAP, ND. *Entailment. The Logic of relevance and necessity, Vol. I*. Princeton: Princeton University Press, 1976.
- [13] ANDERSON, PW. More is Different. *Science*, 1972, 177, 393-396.
- [14] ANDERSON, PW; ARROW, KJ; PINES, D (Eds.). *The Economy as an Evolving Complex System*. Sante Fe, New Mexico: Santa Fe Institute, 1988. Studies in the Sciences of Complexity.
- [15] ANGELINE, PJ; KINNEAR, KE (eds.). *Advances in Genetic Programming, Volume 2*. Cambridge, MA: MIT Press, 1994.
- [16] APOSTEL, L. Towards the Formal Study of Models in the Non-Formal Sciences. *Synthese*, 1960, 12, 125-161.
- [17] ARBIB, MA (Ed.). *Algebraic Theory of Machines, Languages and Semigroups*. New York: Academic Press, 1968.

- [18] ARBIB, MA; MANES, EG. Foundations of system theory: decomposable systems. *Automatica*, 1974, 10, 285-302.
- [19] ARTHUR, B. Why do things become more complex? *Scientific American*, 1993, 268, 92.
- [20] ARTHUR, B. Inductive Reasoning and Bounded Rationality. *American Economic Association Papers and Proceedings*, 1994, 84, 406-411.
- [21] ARTHUR, WB. On the Evolution of Complexity. In COWNAN, G; PINES, D; MELTZER, D (Eds). *Complexity: Metaphors, Models and Reality*. Reading, MA: Addison-Wesley, 1994, 65-78.
- [22] ARTHUR, WB et al. (1996). Asset pricing under endogenous expectations in an artificial stock market. University of Wisconsin-Madison *SSRI report 9625*, 1180 Observatory Drive, Madison, WI 53706, USA
- [23] ASHBY, WR. *Introduction to Cybernetics*. New York: Methuen, 1964.
- [24] ASHBY, WR. Some Peculiarities of Complex Systems. *Cybernetic Medicine*, 1973, 9, 1-7.
- [25] ASHBY, WR; GARDNER, M. Connectance of Large, Dynamic Cybernetic Systems: Critical Values for Stability. *Nature*, 1970, 228, 784.
- [26] ATLAN, H. Natural Complexity and the Self-Creation of Meaning. In AIDA, et al (Ed). *The Science and Praxis of Complexity*. Tokyo: United Nations University, 1984, 173-192.
- [27] ATLAN, H. The Self-Creation of Meaning. *Physica Scripta*, 1987, 36, 563.
- [28] ATMANSPACHER, H; KURTHS, J; SCHEINGRABER, H; et al. Complexity and meaning in nonlinear dynamical systems. *Open systems and Information Dynamics*, 1992, 1, 269-289.
- [29] AUERBACH, D. Dynamical Complexity of Strange Sets. In ABRAHAM, NB (Ed). *Measures of Complexity and Chaos*. New York: Plenum Press, 1989, 203-207.
- [30] AUGER, PM. Hierarchically Organized Systems - The Response to Complexity. *Journal of Scientific & Industrial Research*, 1992, 51, 725-735.
- [31] AUSIELLO, G. Abstract Computational Complexity and Cycling Computations. *Journal of Computer and System Sciences*, 1971, 5, 118-128.
- [32] AYER, A. J. *Language, Truth and Logic*. London: Victor Gollanz, 1936.
- [33] BADIO, R. Unfolding Complexity in Nonlinear Dynamical Systems. In ABRAHAM, NB et al (Ed). *Measures of Complexity and Chaos*. New York: Plenum Press, 1989, 313-325.
- [34] BADIO, R. Complexity and Unpredictable Scaling of Hierarchical Structures. In BOUNTIS, T (Ed). *Chaotic Dynamics, Theory and Practice*. New York and London: Plenum Press, 1992, 1-19.
- [35] BADIO, R; FINARDI, M; BROGGI, G. Recursive symbolic Encoding and Hierarchical Description of Complex Systems. In ATMANSPACHER, H; SCHEINGRABER, H (Eds). *Information Dynamics*. New York: Plenum Press, 1991, 35-43.
- [36] BADIO, R; POLITI, A. *Complexity, Hierarchical Structures and Scaling in Physics*. Cambridge: Cambridge University Press, 1993.
- [37] BAIL, WG; ZELKOWITZ, MV. Program complexity using hierarchical abstract computers. *Journal of Computer Languages*, 1988, 13, 109-123.
- [38] BAK, P; PACZUSKI, M. Complexity, contingency and criticality. *Proceeding of the National Academy of Science USA*, 1995, 92, 6689-6696.
- [39] BAKER, GL. *Chaotic Dynamics*. Cambridge: Cambridge University Press, 1990.
- [40] BALCAZAR, JL. *Structural Complexity*, Vol I. Berlin: Springer-Verlag, 1990.
- [41] BALCAZAR, JL. *Structural Complexity*, Vol II. Berlin: Springer-Verlag, 1990.

- [42] BALDWIN, M. *Portraits of Complexity*. Ohio: Battelle Memorial Institute, 1975. Battelle Monographs.
- [43] BANKS, JF; SUNDARAM, RK. Repeated Games, Finite Automata and Complexity. *Games and Economic Behaviour*, 1990, 2, 97-117.
- [44] BANZHAF, W. Self-Replicating Sequences of Binary Numbers: The Build-up of Complexity. *Complex Systems*, 1994, 8, 215-225.
- [45] BAR-HILLEL, Y; KASHER, Y; SHAMIR, E. Measures of Syntactic Complexity. In BOOTH, AD (Ed). *Machine Translation*. Amsterdam: North-Holland, 1967, 30-50.
- [46] BARTLETT, S (Ed.). *Reflexivity: A source book in self-reference*. New York: Elsevier Science, 1992.
- [47] BATES, JE; SHEPARD, HK. Measuring Complexity Using Information Fluctuation. *Physics Letters A*, 1993, 172, 416-425.
- [48] BEATTY, MJ; PAYNE, SK. Listening Comprehension as a Function of Cognitive Complexity. *Communication Monographs*, 1984, 51, 85-89.
- [49] BECHIR, A; KAMINSKA, B. CYCLOGEN: Automatic, Functional-Level Test Generator Based on the Cyclomatic Complexity Measure and on the ROBDD Representation. *IEEE Transactions on Circuits and Systems - II: Analogue and Digital Signal Processing*, 1995, 42, 446-452.
- [50] BEDNARZ, J. Complexity and Intersubjectivity. *Human Studies*, 1984, 7, 55-70.
- [51] BEER, S. Managing Modern Complexity. *Futures*, 1970, 2, 245-257.
- [52] BEJA, A. Finite and Infinite Complexity in Axioms of Rational Choice or Sen's Characterization of Preference-Compatibility Cannot Be Improved. *Journal of Economic Theory*, 1989, 49, 339-346.
- [53] BENNETT, CH. Dissipation, Information, Computational Complexity and the Definition of Organization. In PINES, D (Ed). *Emerging Syntheses in Science*. Redwood city, CA: Addison-Wesley, 1985, 215-233.
- [54] BENNETT, CH. On the Nature and Origin of complexity in Discrete, Homogenous, Locally-Interacting Systems. *Foundations of Physics*, 1986, 16, 585-592.
- [55] BENNETT, CH. Information, Dissipation, and the Definition of Organization. In PINES, D (Ed). *Emerging Syntheses of Science*. Redwood City, California: Addison-Wesley, 1987, 256.
- [56] BENNETT, CH. Logical Depth and Physical Complexity. In HERKEN, R (Ed). *The Universal Turing Machine, A Half-Century Survey*. Oxford: Oxford University Press, 1988, 227-257.
- [57] BENNETT, CH. How to Define Complexity in Physics, and Why. In ZUREK, WH (Ed). *Complexity, Entropy and the Physics of Information*. Redwood City California: Addison-Wesley, 1990, 137-148.
- [58] BERLEKAMP, ER; CONWAY, JH; GUY, RK. *Winning Ways for Your Mathematical Plays*. New York: Academic Press, 1983.
- [59] BERLYNE, DE. Novelty, Complexity and hedonic value. *Perception and Psychophysics*, 1970, 8, 279-286.
- [60] BERRY, BJL. Comprehending Complexity. *Urban Geography*, 1994, 15, 695-697.
- [61] BETH, E. *Natuurphilosophie*. Gorinchen: Noorduyn, 1948.
- [62] BETH, E. Towards an Up-to-date Philosophy of the Natural Sciences. *Methodos*, 1949, 1, 178-185.

- [63] BETH, E. Semantics of Physical Theories. In FREUDANTHAL, H. (Ed.). *The Concept and the Role of the Model in Mathematics and Natural and Social Sciences*. Dordrecht: Reidel, 1961, 48-51.
- [64] BLACK, M. *Models and Metaphors*. Studies in Language and Philosophy. Ithaca, CA: Cornell University Press, 1962.
- [65] BLUM, L; BLUM, M. Toward a Mathematical Theory of Inductive Inference. *Information and Control*, 1975, 35, 125-155.
- [66] BLUM, L; CUCKER, F; SHUB, M; et al. Complexity and Real Computation: A Manifesto. *International Journal of Bifurcation and Chaos*, 1996, 6, 3-26.
- [67] BLUM, M. A Machine Independent Theory of the Complexity of Recursive Functions. *Journal of the Association of Computing Machinery*, 1967, 14, 322.
- [68] BOGEN, J; WOODWARD, J. Saving the Phenomena. *The Philosophical Review*, 1988, 98,;303-352.
- [69] BOHANEK, M; BRATKO, I. Trading Accuracy for Simplicity in Decision Trees. *Machine Learning*, 1994, 15, 223-250.
- [70] BÖRGER, E. *Computability, Complexity, Logic*. Amsterdam: North Holland, 1989.
- [71] BONNER, JT. *The Evolution of Complexity*. Princeton, NJ: Princeton University Press, 1988.
- [72] BOWDEN, K. Hierarchical tearing: an efficient holographic algorithm for system decomposition. *International Journal of General Systems*, 1994, 23, 23-37.
- [73] BRAITHWAITE, R. B. Models in the Empirical Sciences. In NAGEL, E; SUPPES, P; TARSKI, A (eds.). *Logic Methodology and the Philosophy of Science: Proceedings of the 1960 International Congress*. Stanford, CA: Stanford University Press, 1962, 224-239.
- [74] BREMERMANN, H. Algorithms, complexity, transcomputability and the analysis of systems. *Proceedings of the Fifth Congress of the Deutsche Gesellschaft für Kybernetik*, Munich (eds) Reidel,W, Handler,W Spreng,M, Oldenbourg, 1974.
- [75] BREMERMANN, H. Complexity of Automata, Brains and Behaviour. *Lecture Notes in BioMathematics*, 1974, 4, 304-331.
- [76] BREWER, GD. Analysis of Complex Systems: An experiment and its implications for policy making. Rand Corporation Report P-4951, Santa Monica, CA, 1973.
- [77] BROOKS, RJ; TOBIAS, AM. Choosing the best model: level of detail, complexity and model performance. *Mathematical and Computer Modelling*, 1996, 24, 1-14.
- [78] BUNGE, M. The Weight of Simplicity in the Construction and Assaying of Scientific Theories. *Philosophy of Science*, 1961, 28, 120-149.
- [79] BUNGE, M. *The Myth of Simplicity*. Englewood Cliffs: Prentice-Hall, 1963.
- [80] BURGIN, MS. Generalised Kolmogorov Complexity and other dual Complexity Measures. *Cybernetics*, 1990, 26, 481-491.
- [81] BURLESON, BR et al. More Evidence that Cognitive Complexity is Not Loquacity. *Communication Quarterly*, 34, 317-328
- [82] BURTON, M; DIRKIS, M. Defining Legislative Complexity - A Case study: the tax law improvement project. *University of Tasmania Law Review*, 1996, 14, 198-215.
- [83] CARNEIRO, RL. The Evolution of Complexity in Human Societies and its Mathematical Expression. *International Journal of Comparative Sociology*, 1987, 28, 111-128.
- [84] CARTWRIGHT, N. *How the Laws of Physics Lie*. Oxford: Oxford University Press, 1983.
- [85] CARTWRIGHT, N. Models: The Blueprints for Laws. *Philosophy of Science*, 1997, 64(proc), S292-S303.

- [86] CASTI, JL. Complexity, Connectivity and Resilience in Complex Ecosystems. IFAC Symposium on Bio- and Ecosystems, Liepzig, Germany, 1977.
- [87] CASTI, JL. *Connectivity, Complexity and Catastrophe*. NY: John Wiley, 1979.
- [88] CASTI, JL. On System Complexity: Identification, Measurement and Management. In CASTI, J; KARLQUIST, A (Eds). *Complexity Language and Life: Mathematical Approaches*. Berlin: Springer, 1986, 146-173.
- [89] CASTI, JL. *The simply complex: trendy buzzword or emerging new science?* Bulletin of the Santa Fe Institute, 1992, 7, 10-13.
- [90] CASTI, JL. What if. *New Scientist*, 1996, 151/2038, 36-40.
- [91] CASTI, JL; *Complexification*. London: Abacus, 1994.
- [92] CASTI, JL; KARQUIST, A (Eds.). *Complexity, Language and Life: Mathematical Models*. Berlin: Springer-Verlag, 1985.
- [93] CATTRELL, A. Emergent Properties of Complex Systems. In WESTON-SMITH (Ed). *The Encyclopedia of Ignorance*. Oxford: Pergammon, 1977, 129-135.
- [94] CAVALLO, RE; KLIR, GJ. Reconstructability Analysis: Overview and Bibliography. *International Journal of General Systems*, 1981, 7, 1-6.
- [95] CECI, SJ; LIKER,JK. A Day at the Races: A Study of IQ, expertise and cognitive Complexity, *Journal of Experimental Psychology (General)*, 115, 255-266.
- [96] CELLUCI, C. Proof Theory and Complexity. *Synthese*, 1985, 62, 173-189.
- [97] CHA, S; CHUNG, IS; KWON, YR. Complexity Measures for Concurrent Programs based on Information Theoretic Metrics. *Information Processing Letters*, 1993, 46, 43-50.
- [98] CHADWICK, GF. The limits of the plannable: stability and complexity in planning and planned systems. *Environment and Planning A*, 1977, 9, 1189-1192.
- [99] CHAITIN, GJ. On the Length of Programs for Computing Finite Binary Sequences. *Journal of the Association of Computing Machinery*, 1966, 13, 547-569.
- [100] CHAITIN, GJ. Algorithmic Information Theory. *IBM Journal of Research and Development*, 1977, 21, 350-9,496.
- [101] CHAITIN, GJ. *Algorithmic Information Theory*. Cambridge: Cambridge University Press, 1987.
- [102] CHAITIN, GJ. Randomness and Complexity in Pure Mathematics. *International Journal of Bifurcation and Chaos*, 1994, 4, 3-15.
- [103] CHAO, Y. R. Models in Linguistics and Models in General. In NAGEL, E; SUPPES, P; TARSKI, A (eds.). *Logic Methodology and the Philosophy of Science: Proceedings of the 1960 International Congress*. Stanford, CA: Stanford University Press, 1962, 558-566.
- [104] CHAPMAN, GP. The Epistemology of Complexity and some Reflections on the Symposium. In AIDA, et al (Ed). *The Science and Praxis of Complexity*. Tokyo: United Nations University, 1984, 357-374.
- [105] CHERNIAVSKY, JC; SMITH, CH. On Weyuker Axioms for Software Complexity Measures. *IEEE Transactions on Software Engineering*, 1991, 17, 636-638.
- [106] CLARKE, S; TOBIAS, AM. Complexity in Corporate Modelling: A Review. *Business History*, 1995, 37, 17-44.
- [107] COHEN, DE; MADLENER, K; OTTO, F. Separating the Intrinsic Complexity and the Derivational Complexity of the Word Problem for Finitely Presented Groups. *Mathematical Logic Quarterly*, 1993, 39, 143-157.
- [108] COHEN, PR. *Heuristic Reasoning about Uncertainty: An Artificial Intelligence Approach*. Boston, MA: Pitman, 1985.

- [109] COLLINS, AW. On the Question 'Do Numbers Exist?'. *Philosophical Quarterly*, 1998, 48, 23-36.
- [110] CONANT, RC. Detecting subsystems of a complex system. *IEEE Transactions on Systems, Man and Cybernetics*, 1972, 2, 550-553.
- [111] CONKLIN, D; WITTEN, IH. Complexity-Based Induction. *Machine Learning*, 1994, 16, 203.
- [112] COOPER, C. Complexity in C3I Systems. *Complexity International*, 1993, 1, 7.
- [113] CORBIT, JD; GARBARY, DJ. Fractal dimension as a quantitative measure of complexity in plant development. *Proceedings of the Royal Society of London B*, 1995, 262, 1-6.
- [114] CORNACCHIO, JV. Maximum Entropy Complexity Measures. *International Journal of General Systems*, 1977, 3, 267-271.
- [115] CORNACCHIO, JV. System Complexity - A Bibliography. *International Journal of General Systems*, 1977, 3, 267-271.
- [116] COTTRELL, A. Emergent Properties of Complex Systems. In DUNCAN, R; WESTON-SMITH, M (Eds). *Encyclopedia of Ignorance*. Oxford: Pergamon, 1977, 129-135.
- [117] COVENY, P; R. HIGHFIELD. *Frontiers of Complexity: the search for order in a chaotic world*. London: Faber and Faber, 1995.
- [118] COVER, TM. Kolmogorov Complexity, Data Compression, and Inference. In DURAND, H; DI LULLO, M; SINCLAIR, C (Eds). *The Impact of Data Processing Techniques on Communications*. Dordrecht: Nijhoff, 1983, 23-33.
- [119] CRUTCHFIELD, JP. Reconstructing Language Hierarchies. In ATMANSPACHER, H; SCHEINGRABER, H (Eds). *Information Dynamics*. New York: Plenum Press, 1991, 45-60.
- [120] CRUTCHFIELD, JP. Knowledge and Meaning... Chaos and Complexity. In LAM, L; NARODITSKY, V (Eds). *Modeling Complex Phenomena*. Berlin: Springer, 1992, 66-101.
- [121] CRUTCHFIELD, JP. Observing Complexity and The Complexity of Observation. In ATMANSPACHER, H (Ed). *Inside versus Outside*. Berlin: Springer, 1993, 235-272.
- [122] CRUTCHFIELD, JP. The Calculi of Emergence: Computation, Dynamics and Induction. *Physica D*, 1994, 75, 11-54.
- [123] CRUTCHFIELD, JP; YOUNG, K. Inferring Statistical Complexity. *Physics Review Letters*, 1989, 63, 105.
- [124] CUHROV, IP. On the Number of DNF Minimal Relatively Arbitrary Measures of Complexity. *Lecture Notes in Computer Science*, 1987, 278, 92-94.
- [125] CURTIS, B; SHEPPARD, SB; MILLIMAN, P; et al. Measuring the psychological complexity of software maintenance tasks with the Halstead and McCabe metrics. *IEEE Transactions on Software Engineering*, 1979, 5, 96-104.
- [126] CUSHING, J. T. Models and Methodologies in Current Theoretical High-Energy Physics. *Synthese*, 1982, 50, 5-101.
- [127] CUSHING, J. *Quantum Mechanics: Historical Contingency and Copenhagen Hegemony*. Chicago: Chicago University Press, 1994.
- [128] CUTLAND, NJ. *Computability*. Cambridge: Cambridge University Press, 1980.
- [129] DACOSTA, N. C. A; FRENCH, S. The Model-Theoretic Approach in the Philosophy of Science. *Philosophy of Science*, 1990, 57, 248-265.
- [130] D'ALESSANDRO, G; POLITI, A. Hierarchical approach to complexity with applications to dynamic systems. *Physics Review Letters*, 1990, 64, 1609-1612.

- [131] DAMPNEY, CNG; JOHNSON, SJ; DEUBLE, P. Taming Large Complex Information Systems. *Complex Systems* 92, Australia National University, 1993.
- [132] DANZIN, A. The Pervasiveness of Complexity: Common trends, New Paradigms, and Research Orientations. In AIDA, et al (Ed). *The Science and Praxis of Complexity*. Tokyo: United Nations University, 1984, 69-80.
- [133] DARLEY, V. Emergent Phenomena and Complexity. *Alife IV*, also available electronically at URL: <http://www.cogs.susx.ac.uk/users/ezequiel/alife-page/complexity.html>, 1994.
- [134] DARLEY, VM; KAUFFMAN, SA (forthcoming). Natural Rationality. In ARTHUR, WB et al. (eds) *The Economy as a Complex Evolving System*, Volume II. Reading MA: Addison-Wesley, forthcoming.
- [135] DAVIS, JS; LEBLANC, RJ. A Study of the Applicability of Complexity Measures. *IEEE Transactions on Software Engineering*, 1989, 14, 1366-1372.
- [136] DAWKINS, R. *The Blind Watchmaker*. London: Penguin, 1986.
- [137] DAWKINS, R. The evolution of evolvability. In C.G.LANGTON (Ed.). *Artificial Life*. Reading, MA: Addison-Wesley, 1989.
- [138] DE LUCA, A. Complexity and Information Theory. In G.LONGO (Ed). *Coding and Complexity*. Berlin: Springer-Verlag, 1975, 207-270.
- [139] DIAMOND, AH; MCKINSEY, JCC. Algebras and their subalgebras. *Bulletin of the American Mathematical Society*, 1947, 53, 959-962.
- [140] DOMOTOR, Z. Measurement: empiricist and realist views. In SAVAGE, CW; EHRLICH, P (eds.). *Philosophical and Foundations Issues in Measurement Theory*, Hillsdale, NJ: Lawrence Erlbaum, 1992, 195-221.
- [141] DRAPER, NR; SMITH, H. *Applied Regression Analysis*. New York: John Wiley, 1981.
- [142] DUNNE, PE. *Complexity of Boolean Networks*. London: Academic Press, 1988.
- [143] DUHEM, P. *The Aim and Structure of Physical Theory*. Translated by WIENER, PP, (1954) Princeton: Princeton University Press, 1914.
- [144] DUSSAUCHOY, RL. Generalized Information theory and the Decomposability of Systems. *International Journal of General Systems*, 1982, 9, 13-36.
- [145] DYKE, C. *The Evolutionary Dynamics of Complex Systems*. New York: OUP, 1988.
- [146] EBELING, W; JIMENEZ-MONTANO, MA. On Grammars, Complexity and Information Measures of Biological Macromolecules. *Mathematical Bioscience*, 1980, 52, 53-71.
- [147] EDMONDS, B. What is Complexity? - The philosophy of complexity per se with application to some examples in evolution, in F. Heylighen & D.Aerts (eds.): *The Evolution of Complexity*, Kluwer, Dordrecht, 1998.
- [148] EDMONDS, B. Modelling Bounded Rationality using Evolutionary Techniques. AISB Workshop on Evolutionary Computing, Manchester, 1997. *Lecture Notes in Computer Science*, 1997, 1305, 31-42.
- [149] EDMONDS, B. From Complexity to Agent Modelling and Back Again - some implications for economics. Workshop on Economics and the Sciences of Complexity, Namur, Brussels, May 1997.
- [150] EDMONDS, B. Complexity and Scientific Modelling. 20<sup>th</sup> International Wittgenstein Symposium, Kirchberg am Wechsel, Austria, August 1997.
- [151] EDMONDS, B; MOSS, SJ; WALLIS, S. Logic, Reasoning and a Programming Language for Simulating Economic and Business Processes with Artificial Intelligent Agents. In

- EIN-DOR, P. (ed.): *Artificial Intelligence in Economics and Management*. Boston: Kluwer Academic Publishers, 221-230, 1996.
- [152] EDMONDS, B. Capturing Social Embeddedness: a Constructivist Approach. *Adaptive Behavior*, forthcoming.
- [153] EHRENFEUCHT, A; ZEILGER, P. Complexity Measures for Regular Expressions. 6th Annual ACM Symposium on the Theory of Computation, Seattle, WA, 1974, 1974.
- [154] EPSTEIN, RL. *The Semantic Foundations of Logic, Vol. 1: Propositional Logics*. Dordrecht: Kluwer, 1990.
- [155] FELDMAN, JA. Some decidability results on grammatical inference and complexity. *Information and Control*, 1972, 20, 244-262.
- [156] FERDINAND, AE. Complex systems are those that work even with maximum entropy. *International Journal of General Systems*, 1974, 1, 19-33.
- [157] FLOOD, RL. Complexity: A definition by construction of a conceptual framework. *Systems Research*, 1987, 4, 177-185.
- [158] FLOOD, RL; CARSON, ER. *Dealing with Complexity*. New York: Plenum Press, 1988.
- [159] FOSTER, MH; MARTIN, ML (Eds.). *Probability, Confirmation and Simplicity*. New York: Odyssey Press, 1966.
- [160] FRASER, AM. Measuring Complexity in Terms of Mutual Information. In ABRAHAM, NB (Ed). *Measures of Complexity and Chaos*. New York: Plenum Press, 1989, 117-119.
- [161] VAN FRASSEN, BC. *The Scientific Image*. Oxford: Clarendon, 1980.
- [162] FREGE, G. *Begriffsschrift*. Halle: Nebert, 1879.
- [163] FUTIA, C. The Complexity of Economic Decision Rules. *Journal of Mathematical Economics*, 1975, 4, 289-299.
- [164] GAINES, BR. On the Complexity of Casual Models. *IEEE Transactions on Systems, Man and Cybernetics*, 1975, 6, 56-59.
- [165] GAINES, BR. System Identification, Approximation and Complexity. *International Journal of General Systems*, 1976, 3, 145-174.
- [166] GALBRAITH, J. *Designing Complex Organisations*. Reading, MA: Addison-Wesley, 1973.
- [167] GALLO, G; LONGO, G; PALLOTTINO, S; et al. Directed Hypergraphs and Applications. *Discrete Applied Mathematics*, 1993, 42, 177-201.
- [168] GELL-MANN, M. *The Quark and the Jaguar - adventures in the simple and the complex*. London: Little, Brown and Co., 1994.
- [169] GELL-MANN, M. What is Complexity? *Complexity*, 1995, 1, 16-19.
- [170] GELL-MANN, M; LLOYD, S. Information Measures, Effective Complexity and Total Information. *Complexity*, 1996, 2, 44-52
- [171] GEORGE, B. Tests for System Complexity. *International Journal of General Systems*, 1977, 3, 253-257.
- [172] GEORGE, DAR. Chaos and Complexity in Economics: Review Article. *Journal of Economic Surveys*, 1990, 4, 397-404.
- [173] GIARINI, O. The Consequences of Complexity in Economics: Vulnerability, Risk, and Rigidity Factors in Supply. In AIDA, et al (Ed). *The Science and Praxis of Complexity*. Tokyo: United Nations University, 1984, 133-145.

- [174] GIBSON, E; SCHUTZE, CT; SALOMON, A. The Relationship Between the Frequency and the Processing Complexity of Linguistic Structure. *Journal of Psycholinguistic Research*, 1996, 25, 59-92.
- [175] GIERE, RN. Constructive Realism. In CHURCHLAND, PM; HOOKER, CW (eds.). *Images of Science*, Chicago: University of Chicago Press, 1985, 75-98.
- [176] GIERE, RN. The Cognitive Structure of Scientific Theories. *Philosophy of Science*, 1994, 61, 276-296.
- [177] GIRARD, JY. Linear Logic. *Theoretical Computer Science*, 1987, 50, 1-102.
- [178] GIRARD, JY. *Proof Theory and Logical Complexity*. Naples: Bibliopolis, 1987.
- [179] GLUCK, R; KLIMOV, A. Reduction of Language Hierarchies. 14th International Congress on Cybernetics, Namur, Belgium, 1995, 1995.
- [180] GÖDEL, K. On the length of proofs. *Ergebnisse eines mathematischen Kolloquiums*, 1936, 7, 23-24.
- [181] GODFREY-SMITH, P. *Complexity and the Function of Mind in Nature*. Cambridge: Cambridge University Press, 1996.
- [182] GOGUEN, JA. Complexity of Hierarchically Organised Systems. *International Journal of General Systems*, 1977, 3(4), 233-251.
- [183] GOLAY, MW; SEONG, PH; MANNO, VP. A Measure of the Difficulty of System Diagnosis and its Relationship to Complexity. *International Journal of General Systems*, 1989, 16, 1-23.
- [184] GOLDMAN, SA; KEARNS, MJ; SCHAPIRE, RE. On the Sample Complexity of Weak Learning. *Information and computation*, 1995, 117, 276.
- [185] GOOD, IJ. Corroboration, Explanation, Evolving Probability and a Sharpened Razor. *British Journal for the Philosophy of Science*, 1969, 19, 123-43.
- [186] GOODMAN, N. *The Structure of Appearance*. Indianapolis: Bobbs-Merrill, 1966.
- [187] GOLDSMITH, RE; NUGENT, N. Innovativeness and Cognitive Complexity, *Psychological Reports*, 1984, 55, 431-438.
- [188] GORODKIN, J; SORENSEN, A; WINTHER, O. Neural Networks and Cellular Automata Complexity. *Complex Systems*, 1993, 7, 1-23.
- [189] GOTTINGER, HW. Complexity and Information Technology in Dynamic Systems. *Kybernetes*, 1975, 4, 129-141.
- [190] GOTTINGER, HW. Complexity and Dynamics: Applications of Dynamic Systems Theory. *IEEE Transactions on Systems, Man and Cybernetics*, 1976, 6, 867-873.
- [191] GOTTINGER, HW. Choice and Complexity. *Mathematical Social Sciences*, 1987, 14, 1-17.
- [192] GOTTINGER, HW; LEINFELLNER, W (Eds.). *Decision Theory and Social Ethics: Issues in Social Choice*. Dordrecht: Reidel, 1978.
- [193] GRASSBERGER, P. Towards a quantitative theory of self-generated complexity. *International Journal of Theoretical Physics*, 1986, 25(9), 907-938.
- [194] GRASSBERGER, P. Problems in Quantifying Self-organized complexity. *Helvetica Physica Acta*, 1989, 62, 498-511.
- [195] GRASSBERGER, P. Information and Complexity Measures in Dynamical Systems. In ATMANSPACHER, H; SCHEINGRABER, H (Eds.). *Information Dynamics*. New York: Plenum Press, 1991, 15-33.
- [196] GREEN, DG. Syntactic modelling and simulation. *Simulation*, 1990, 54, 281-286.

- [197] GREEN, DG. Emergent Behaviour in Biological Systems. In GREEN, DG; BOSSOMAIER, TJ (Eds). *Complex Systems - From Biology to Computation*. Amsterdam: OS Press, 1993, 24-35.
- [198] GREENBERG, WJ. A Theory of Complexes. Preprint, department of English, college of humanities, University of Puerto Rico at Rio Piedras, 1990.
- [199] GROENEWOLD, H. J. The model in physics. *Synthese*, 1960, 12, 222-227.
- [200] GRUNBERG, E. "Complexity" and "Open Systems" in Economic Discourse. *Journal of Economic Issues*, 1978, 12, 541-560.
- [201] GUNSTEREN, W. F; BERENDSEN, H. J. C. Computational Simulation of Molecular Dynamics: Methodology, Applications and Perspectives in Chemistry. *Angewandte Chemie – International Edition in English*, 1990, 29, 992-1023.
- [202] GUNTHER, R; SHAPIRO, B; WAGNER, P. Complex Systems, complexity measures, grammars and model inferring. *Chaos, Solitons and Fractals*, 1994, 4, 635-651.
- [203] GUSEV, VD; KULICHKOV, VA; CHUPKHINA, OM. Genome Complexity Analysis 1: Complexity Measures and the Classification of Structural Features. *Molecular Biology*, 1991, 25, 669-677.
- [204] HAGESETH, JA. Relationships among cognitive complexity variables, *Psychological Reports*, 1983, 52, 473-474.
- [205] HAKEN, H. *Information and Self-Organisation: a macroscopic approach to complex systems*. Berlin: Springer-Verlag, 1988. Springer Series in Synergetics.
- [206] HALSTEAD, MH. *Elements of Software Science*. New York: Elsevier, 1977.
- [207] HARNAD, S. The symbol grounding problem. *Physica D*, 1981, 42, 335-346.
- [208] HARTMAN, S. Modelling and the Aims of Science. 20<sup>th</sup> International Wittgenstein Symposium, Kirchberg am Wechsel, 1997.
- [209] HAWRNACK, J; ZYGMUNT, J. On the Degree of Complexity of Sentential Logics. *Studia Logica*, 1981, 4, 142-153.
- [210] HAYEK, FA. The Theory of Complex Phenomena. In BUNGE, M (Ed). *The Critical Approach to Science and Philosophy*. London: Collier McMillan, 1964, 332-349.
- [211] HEMMERLING, A. On Genuine Complexity and Kinds of Nondeterminism. *Journal of information processing and cybernetics*, 1995, 30, 77.
- [212] HEMPEL, C. G. *Aspects of Scientific Explanation*. New York: Free Press, 1965.
- [213] HEMPEL, CG; OPPENHEIM, P. Studies in the Logic of Explanation. *Philosophy of Science*, 1948, 15, 135-175.
- [214] HENRY, S; KAFURA, K. Software Structure Metrics based on Information Flow. *IEEE Transactions on Software Engineering*, 1981, 7(5), 510-518.
- [215] HERROELEN, W; VANGLIS, A. On the use of Flow Dominance in Complexity Measures for Facility Layout Problems. *International Journal of Production Research*, 1985, 23, 97-108.
- [216] HESSE, MB. *Models and Analogies in Science*. London: Sheed and Ward, 1963.
- [217] HESSE, MB. Simplicity. In EDWARDS, P (Ed). *Encyclopedia of Philosophy*. New York: Macmillan, 1967, 445-449.
- [218] HEYLIGHEN, F. Relational Closure: a Mathematical Concept for Distinction-making and complexity analysis. In TRAPPL, R (Ed). *Cybernetics and Systems '90*. Singapore: World Scientific, 1990, 335-341.

- [219] HEYLIGHEN, F. Modelling Emergence. ftp papers from Principia Cybernetica server, also for "world Futures: the Journal of General Evolution", special issue on creative evolution, G. Kampis (ed), 1991.
- [220] HEYLIGHEN, F. Self-Organization, Emergence and the Architecture of Complexity. In *Proceedings of the European Congress on System Science*. Paris: AFCET, 1992, 23-32.
- [221] HEYLIGHEN, F. The Growth of Structural and Functional Complexity during Evolution. In HEYLIGHEN, F & AERTS, D (eds.) *The Evolution of Complexity*. Amsterdam: Kluwer Academic Publishers, 1996.
- [222] HILBERT, D.; BERNAYS, P. *Grundlagen der Mathematik*, Vol. 1. Berlin: Springer, 1934.
- [223] HINEGARDNER, R; ENGELBERG, J. Biological Complexity. *Journal of Theoretical Biology*, 1983, 104, 7-20.
- [224] HITCHCOCK, D. Relevance. *Argumentation*, 1992, 6, 251-270.
- [225] HO, JKK; SCULLI, D. System Complexity and the Design of Decision Support Systems. *Systems Practice*, 1995, 8, 505-516.
- [226] HOGG, T; HUBERMAN, BA. Order, Complexity and Disorder. *Mondes en Developpement*, 1986, 54-55.
- [227] HOGG, T; HUBERMAN, BA; JACQUELINE, MM. The stability of ecosystems. *Proceedings of the Royal Society of London B*, 1989, 237, 43-51.
- [228] HOLLAND, J. *Adaption in Natural and Artificial Systems*. Ann Arbor, Michigan: University of Michigan Press, 1975.
- [229] HOLM, HJ. *Complexity in Economic Theory*. Lund, Sweden: University of Lund: Lund Economic Studies, 1993.
- [230] HOOKER, JN; VINAY, V. Branching Rules for Satisfiability. Carnegie Mellon University, Graduate School of Industrial Administration working paper WP 1994-09, 1994.
- [231] HOPCROFT, JE; ULLMAN, JD. *Introduction to Automata Theory, Languages, and Computation*. Reading, Massachusetts: Addison-Wesley, 1979.
- [232] HOPS, JM; SHERIF, JS. Development and application of composite complexity models and a relative complexity metric in a software maintenance environment. *Journal of Systems and Software*, 1995, 31, 157-169.
- [233] HORGAN, J. From Complexity to Perplexity. *Scientific American*, 1995, 272, 74-79.
- [234] HOWATT, JW; BAKER, AL. Rigorous Definition and Analysis of Program Complexity Measures - An example using nesting. *Journal of Systems and Software*, 1989, 10, 139-150.
- [235] HROMKOVIC, J; KARI, J; KARI, L. Some Hierarchies for the Communication Complexity Measures of Cooperating Grammar Systems. *Theoretical Computer Science*, 1994, 127, 123-147.
- [236] HUGHES, R. G. Models and Representation. *Philosophy of Science*, 1997, 64(proc), S325-S336.
- [237] HUBERMAN, BA; HOGG, T. Complexity and Adaption. *Physica D*, 1986, 22, 376-384.
- [238] ITAI, A; MAKOWSKY, JA. Unification as a Complexity Measure for Logic Programming. *Journal of Logic Programming*, 1987, 4, 105-117.
- [239] JACKO, JA; SALVENDY, G. Hierarchical menu design: breadth, depth and task complexity. *Perceptual and Motor Skills*, 1996, 82, 1187-1202.
- [240] JASKOWSKI, S. Trois Contributions au Calcul des Propositions Bivalentes. *Studia Societatis Scientiarum Torunansis*, 1948, 1(1), 1-15.
- [241] JEFFREYS, H. *Scientific Inference*. Cambridge: Cambridge University Press, 1937.

- [242] JOHNSON, J. A Theory of Stars in complex Systems. In CASTI, JL; KARLQUIST, A (Eds). *Complexity Language and Life: Mathematical Approaches*. Berlin: Springer-Verlag, 1986, 21-61.
- [243] KAMPIS, G. Information, Computation and Complexity. In CARVALLO, ME (Ed). *Nature, Cognition and Systems*. Dordrecht: Kluwer, 1988, 313-320.
- [244] KAMPIS, G. *Self-Modifying Systems in Biology and Cognitive Science: A New Framework for Dynamics, Information and Complexity*. Oxford: Pergamon Press, 1991.
- [245] KAMPIS, G; CSANYI, V. Simple Models Do Not Eliminate Complexity from the Real World. *Journal of Theoretical Biology*, 1985, 115, 467-469.
- [246] KAMPIS, G; CSANYI, V. Notes on Order and Complexity. *Journal of Theoretical Biology*, 1987, 124, 111-121.
- [247] KANEKO, K; TSUDA, I. Constructive Complexity and Artificial Reality: An Introduction. *Physica D*, 1994, 75, 1-10.
- [248] KAPLOW, L. A Model of the Optimal Complexity of Rules. National Bureau of Economic Research Working paper No. 3958, 1992.
- [249] KAUFFMAN, SA. *The Origins of Order*. New York: Oxford University Press, 1993.
- [250] KELLY, GA. *The Psychology of Personal Constructs*, New York: Norton, 1955.
- [251] KELLY, JS. Social Choice and Computational Complexity. *Journal of Mathematical Economics*, 1988, 17, 1-8.
- [252] KELMANS, AK. A Comparison of Graphs by their number of spanning trees. *Discrete Mathematics*, 1976, 16, 241-261.
- [253] KEMENY, JG. The Use of Simplicity in Induction. *The Philosophical Review*, 1953, 62, 391-408.
- [254] KEMENY, JG. Two Measures of Complexity. *The Journal of Philosophy*, 1953, 52, 722-733.
- [255] KEMENY, JG. A New Approach to Semantics, Part I. *Journal of Symbolic Logic*, 1956.
- [256] KEMPER, S; RICE, K; CHEN, Y. Complexity metrics and growth curves for measuring grammatical development from five to ten. *First Language*, 1995, 15, 151-166.
- [257] KHALIL, EL. Organizations, Naturalism and Complexity. *Review of Social Economy*, 1995, 53, 393-419.
- [258] KIMELMAN, D; LEBAN, B. Reduction of Visual Complexity in Dynamic Graphs. DIMACS Graph Drawing'94, 1994.
- [259] KINDLMANN, P. Stability v. Complexity in Model Computational Communities. *Lecture Notes in BioMathematics*, 1984, 54, 191-207.
- [260] KINGSNORTH, A. Complexities of Complexity. *Bulletin of the American Schools of Oriental Research*, 1993, 292, 107-120.
- [261] KINNEAR, KE (ed.). *Advances in Genetic Programming*. Cambridge, MA: MIT Press, 1994.
- [262] KLIR, GJ. The Many Faces of Complexity. In AIDA, et al (Ed). *The Science and Praxis of Complexity*. Tokyo: United Nations University, 1984, 81-98.
- [263] KLIR, GJ. *Architecture of Systems Problem Solving*. New York: Plenum Press, 1985.
- [264] KLIR, GJ. Complexity: Some General Observations. *Systems Research*, 1985, 2, 131-140.
- [265] KLIR, GJ; FOLGER, TA. *Fuzzy Sets, Uncertainty and Information*. New Jersey: Prentice Hall, 1988.

- [266] KOLMOGOROV, AN. Three Approaches to the quantitative definition of Information. *Problems of Information Transmission*, 1965, 1, 1-17.
- [267] KOLMOGOROV, AN; TICHOMIROV, WM. e-Entropie und e-Kapazität von Mengen in Funktionalräumen. In *Arbeiten zur Informationstheorie*. Berlin: VEB Deutscher Verlag der Wissenschaften, 1960.
- [268] KOPPEL, M. Complexity, Depth and Sophistication. *Complex Systems*, 1987, 1, 1087-1091.
- [269] KOPPEL, M. Structure. In HERKEN, R (Ed). *The Universal Turing Machine, A Half Century Survey*. Oxford: Oxford University Press, 1988, 435-452.
- [270] KOZA, JR. Genetic Programming: On the Programming of Computers by Natural Selection. Cambridge, MA: MIT Press, 1992.
- [271] KRANTZ, D; LUCE, D; SUPPES, P; TVERSKY, A. *Foundations of Measurement, Vol. I*. New York: Academic Press, 1971.
- [272] KROHN, K; RHODES, JL. Complexity of Finite Semi-Groups. *Annals of Mathematics*, 1968, 88, 128-160.
- [273] KROLL, NEA; KLIMESCH, W. Semantic Memory - Complexity or Connectivity. *Memory and Cognition*, 1992, 20, 192-210.
- [274] KUHN, TS. *The Structure of Scientific Revolutions*. Chicago: Chicago University Press, 1962.
- [275] KUHN, TS. Some Thoughts on Paradigms. In SUPPE, F (ed.). *The Structure of Scientific Theories*. Urbana, IL: University of Urbana Press, 1973, 459-482.
- [276] KYBURG, HE. Measuring Errors of Measurement. In SAVAGE, C. W; EHRLICH, P (eds.). *Philosophical and Foundations Issues in Measurement Theory*, Hillsdale, NJ: Lawrence Erlbaum, 1992, 75-91.
- [277] LAKSHMANAN, KB; JAYAPRAKASH, S; SINHA, PK. Properties of Control-Flow Complexity Measures. *IEEE Transactions on Software Engineering*, 1991, 17, 1289-1295.
- [278] LAZAREV, VI. Complexity and Synthesis of Minimal Logic Circuits using Multiplexers. *Cybernetics*, 1992, 28, 796-799.
- [279] LEMPEL, A; ZIV, J. On the complexity of finite sequences. *IEEE Transactions on Information Theory*, 1976, 22, 75-81.
- [280] LESTEL, D. The metaphors of complexity: the language and cognitive resources of artificial life. *Social Science Information*, 1996, 35, 511-540.
- [281] LEVESQUE, HJ. Logic and the Complexity of Reasoning. *Journal of Philosophical Logic*, 1989, 17, 355-389.
- [282] LEVINE, RD; TRIBUS, M (Eds.). *The Maximum Entropy Principle*. Cambridge, Massachusetts: MIT Press, 1979.
- [283] LEVINS, RD. The Limits of Complexity. In PATTEE, HH (Ed). *Hierarchy Theory - The Challenge of Complex Systems*. New York: George Braziller, 1973, 109-127.
- [284] LI, M; VITANYI, JMB. Statistical Properties of Finite Sequences with High Kolmogorov Complexity. *Mathematical Systems Theory*, 1994, 27, 365-376.
- [285] LI, M; VITANYI, PMB. Philosophical Issues in Kolmogorov Complexity. *Lecture Notes in Computer Science*, 1992, 623, 1-15.
- [286] LI, M; VITANYI, PMB. *An Introduction to Kolmogorov Complexity and Its Applications*. New York: Springer-Verlag, 1993.
- [287] LI, W. On the Relationship between Complexity and Entropy for Markov Chains and Regular Languages. *Complex Systems*, 1991, 5, 381-399.

- [288] LIGGGESMEYER, P. A set of complexity metrics for guiding the software test process. *Software Quality Journal*, 1995, 4, 257-273.
- [289] LINDGREN, K; NORDAHL, MG. Complexity measures and cellular automata. *Complex Systems*, 1988, 2, 409.
- [290] LIPMAN, BL; SRIVASTAVA, S. Informational Requirements and Strategic Complexity in Repeated Games. *Games and Economic Behaviour*, 1990, 2, 273-290.
- [291] LLOYD, S. Physical Measures of Complexity. In JEN, E (Ed). 1989 *Lectures in Complex Systems*. Redwood City, California: Addison-Wesley, 1990, 67-73.
- [292] LLOYD, S; PAGELS. Complexity as Thermodynamic Depth. *Annals of Physics*, 1988, 188, 186-213.
- [293] LÖFGREN, L. Complexity of Descriptions of Systems: A foundational Study. *International Journal of General Systems*, 1974, 3, 197-214.
- [294] LÖFGREN, L. Syntactic and Semantic reduction of Theories. *Journal for Symbolic Logic*, 1976, 41, 268-269.
- [295] LÖFGREN, L. Complexity of Systems. In SINGH, M (Ed). *Systems and Control Encyclopedia*. Oxford: Pergamon, 1987, 704-709.
- [296] LOPEZ, LR; CAUFIELD, LJ. A Principle of Minimum Complexity in Evolution. *Lecture Notes in Computer Science*, 1991, 496, 405-409.
- [297] LØPEZ-RUIZ, R; MANCINI, HL; CALBERT, X. A statistical measure of complexity. *Physics Letters A*, 1995, 209, 321-326.
- [298] LUGOSI, G; ZEGER, K. Concept Learning Using Complexity Regularization. *IEEE Transactions on Information Theory*, 1996, 42, 48-54.
- [299] LUHMANN, N. Complexity and Meaning. In AIDA ET AL. (Ed). *The Science and Praxis of Complexity*. Tokyo: United Nations University, 1985, 99-104.
- [300] LUKASIEWICZ, J. The Shortest Axiom of the Implicational Calculus of Propositions. *Proceedings of the Royal Irish Academy*, 1948, 52, 25-33.
- [301] LUPTON, P. Simplicity and Misrepresentation. Available electronically at URL: <http://cogsci.l.chiba-u.ac.jp/IPPE/PhilofMind/Lupton.SimplicityandMisrepresentation/index.html>, 1994.
- [302] LYON, ML. Complexity and Emergence: The seduction and reduction of non-linear models in the social sciences. *Complex Systems* 92, Australia National University, 1992.
- [303] MACHLIN, R. The Complex Behaviour of Simple Machines. *Physica D*, 1990, 42, 85-98.
- [304] MADDOX, J. Complicated Measures of Complexity. *Nature*, 1990, 344, 705.
- [305] MAKINEN, E. 2 Complexity Measures for Context-free Languages. *International Journal of Computer Mathematics*, 1988, 1, 29-34.
- [306] MAKLAD, MS; NICHOLS, ST. A New Approach to Model Structure Discrimination. *IEEE Transactions on Systems, Man and Cybernetics*, 1980, 10, 78-84.
- [307] MARCUS, MJ. The Theory of Connecting Networks and their Complexity: A Review. *Proceedings of the IEEE*, 1977, 65, 1263-1271.
- [308] MARGALEF, R. Ecosystems: Diversity and Connectivity as measurable components of their complication. In AIDA, et al (Ed). *The Science and Praxis of Complexity*. Tokyo: United Nations University, 1984, 228-244.
- [309] MARTIN-LOF, P. The Definition of Random Sequences. *Information and Control*, 1966, 9, 602-619.
- [310] MARTINEZ, JL. On Complexity and Simplicity as Different Evolutionary Strategies. *Evolutionary Theory*, 1995, 10, 317-320

- [311] MASUCH, M; HUANG, Z. A logical deconstruction of organizational action: formalising J.D. Thompson's 'Organisations in Action' in a multi-agent action logic. *CCSOM working paper 94-120*, Department of Statistics and Methodology, University of Amsterdam, 1994.
- [312] MATA-TOLEDO, RA; GUSTAFSON, DA. A Factor Analysis of Software Complexity Measures. *Journal of Systems and Software*, 1992, 17, 267-273.
- [313] MCCABE. A Complexity Measure. *IEEE Transactions on Software Engineering*, 1976, 2, 308-320.
- [314] MCINTYRE, LC. Complexity and Social Scientific Laws. *Synthese*, 1993, 97, 209-227.
- [315] MCMULLEN, C; SHEARER, J. Prime Implicants, Minimum Cover and the Complexity of Logic Simplification. *IEEE Transactions on Computers*, 1986, 35, 761-762.
- [316] MCSHEA, D. Complexity and Evolution: what everybody knows. *Biology and Philosophy*, 1991, 6(3), 303-3024.
- [317] MCSHEA, D. Meatzoan Complexity and Evolution: is there a trend? *Evolution*, 1996, 50, 477-492
- [318] MEER, K. On the Relations Between Discrete and Continuous Complexity Theory. *Mathematical Logic Quarterly*, 1995, 41, 281-286.
- [319] MEINEL, C. Logic vs. Complexity Theoretic Properties of the Graph Accessibility Problem for Directed Graphs of Bounded Degree. *Information Processing Letters*, 1990, 34, 143-146.
- [320] MEREDITH, CA. On an Extended System of the Propositional Calculus. *Proceedings of the Royal Irish Academy*, 54.
- [321] MEREDITH, CA. Single Axioms for the Systems (C, N), (C, O) and (A, N) of the Two-valued Propositional Calculus. *The Journal of Computing Systems*, 1, paper 10.
- [322] MEYER, AR; RITCHIE, DM. The Complexity of Loop Programs. In *Proceedings of the 22nd National Conference*. Washington DC: Thompson Book Company, 1967, 465-469.
- [323] MIKULECKY, DC. Life, Complexity and the Edge of Chaos: Cognitive Aspects of Communication between cells and other components of living systems. Available electronically at URL: <http://griffin.vcu.edu/complex/mikulecky/rev.htm>, 1995.
- [324] MILLER, DA; ZUCKER, SW. Complexity and Optimality - A Commentary. *Behavioural and Brain Sciences*, 1991, 14, 227.
- [325] MILLER, GA. The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review*, 1956, 63, 81-97.
- [326] MILLER, GA; CHOMSKY, N. Finitary models of language users. In LUCE, RD; BUSH, RR; GALANTER, E (Eds). *Handbook of Mathematical Psychology*. New York: Wiley, 1963, 419-491.
- [327] MITCHELL, S. D. Pragmatic Laws. *Philosophy of Science*, 1997, 64(proc), S468-S479.
- [328] MORGAN, M. S. The Technology of Analogical Models. *Philosophy of Science*, 1996, 64(proc), S304-S314.
- [329] MORGAN, M; MORRISON, M (eds.). *Models as Mediators*. Cambridge: CUP, forthcoming.
- [330] MORIN, E. Complexity. *International Journal of Social Science*, 1974, 26, 583-597.
- [331] MORIN, E. On the Definition of Complexity. In AIDA, EA (Ed). *The Science and Praxis of Complexity*. Tokyo: United Nations University, 1984, 62-68.
- [332] MOSHOWITZ, A. Entropy and the Complexity of Graphs. *Bulletin of Mathematical Biophysics*, 1968, 30, 175-204,225-40.

- [333] MOSS, S; EDMONDS, B. Modelling Economic Learning as Modelling, *Cybernetics and Systems*, 1998, 29, 5-37.
- [334] MOSS, S; EDMONDS, B; GAYLARD, H. Modelling R&D Strategy as a Network Search Problem. Workshop on the Multiple Linkages between Technological Change, Human Capital and the Economy, University 'Tor Vergata' of Rome, March 1996.
- [335] MULHAUSER, GR. To Simulate or Not to Simulate: A Problem of Minimising Functional Logical Depth. *Lecture Notes in Artificial Intelligence*, 1995, 929, 530-543.
- [336] MURPHY, PM; PAZZANO, MJ. Exploring the Decision Forest: An Empirical Investigation of Occam's Razor in Decision Tree Induction. *Journal of Artificial Intelligence Research*, 1994, 1, 257-275.
- [337] NAGEL, E. *The Structure of Science*. New York: Harcourt Brace, 1961.
- [338] NAYLOR, AW. On Decomposition Theory: Generalised Dependence. *IEEE Transactions on Systems, Man and Cybernetics*, 1981, 10, 699-713.
- [339] NELSON, RJ. Structure of Complex Systems. Philosophy of Science Association, 1976, 2, 523-542.
- [340] NIELSEN, ME. H-Comp: a program to calculate information complexity. *Behaviour Research Methods, Instruments and Computers*, 1996, 28, 483-485.
- [341] O'NEAL, MB; EDWARDS, WR. Complexity Measures for Rule-Based Programs. *IEEE Transactions on Knowledge and Data Engineering*, 1994, 6, 669-680.
- [342] O'NEILL, RV; DEANGELIS, DL; WAIDE, JB; et al. *A Hierarchical Concept of Ecosystems*. Princeton, NJ: Princeton University Press, 1986.
- [343] OSHERSON, DN; WEINSTEIN, S. On Advancing Simple Hypothesis. *Philosophy of Science*. 1990, 57, 266-277.
- [344] PAKKAN, M; AKMAN, V. Issues in Common Sense Set-theory. *Artificial Intelligence Review*, 1994, 8(4), 279-308.
- [345] PAPENTIN, F. On Order and Complexity. *Journal of Theoretical Biology*, 1980, 87, 1-456.
- [346] PATTEE, H (Ed.). *Hierarchy Theory, the Challenge of Complex Systems*. New York: Braziller, 1973.
- [347] PATTEE, H. Dynamic and Linguistic Modes of Complex Systems. *International Journal of General Systems*, 1977, 3, 259-266.
- [348] PATTEE, H. Evolving self-reference: matter, symbols and semantic closure. *Communication and Cognition - Artificial Intelligence*, 1995, 12, 9-27.
- [349] PEARL, J. On the Connection Between the Complexity and Credibility of Inferred Models. *International Journal of General Systems*, 1978, 4, 255-264.
- [350] PELITI, L; VULPIANI, A (Eds.). *Measures of Complexity*. Berlin: Springer-Verlag, 1988.
- [351] PERLIS, A; SAYWARD, F; SHAW, M (Eds.). *Software Metrics*. Cambridge (MA): MIT Press, 1981.
- [352] PHILLIPS, W and T, S. Systems Thinking and the Quality of Life. *Proceedings of the Society for General Systems Research Annual Meeting*, 1975.
- [353] PIMM, S. The complexity and stability of ecosystems. *Nature*, 1984, 307, 321-326.
- [354] PINCUS, S. Approximate entropy (ApEn) as a complexity measure. *Chaos*, 1995, 5, 110-117.
- [355] PIPPENGER, N. Complexity Theory. *Scientific American*, 1978, 239, 90-100.
- [356] PIPPENGER, N. Invariance of Complexity Measures for Networks with Unreliable Gates. *Journal of the Association of Computing Machinery*, 1989, 36, 531-539.

- [357] POON, L; GREBOGI, C. Controlling Complexity. *Physical Review Letters*, 1995, 75, 4023-4026.
- [358] POPPER, KR. *Logic of Scientific Discovery*. London: Hutchinson, 1968.
- [359] PRATHER, RE. An axiomatic theory of software complexity research. *The Computer Journal*, 1984, 27, 340-346.
- [360] PRIBRAM, KH. Complexity and Causality. In AIDA, S, et al (Ed). *The Science and Praxis of Complexity*. Tokyo: United Nations University, 1984, 119-132.
- [361] PRIGOGINE, I; STENGERS, I. *Order out of Chaos - Man's new dialogue with nature*. Toronto: Bantam Books, 1984.
- [362] PRIOR, AN. *Formal Logic*. London: Oxford University Press, 1962.
- [363] PUTNAM, H. What Theories are Not. In NAGEL, E; SUPPES, P; TARSKI, A (eds.). *Logic Methodology and the Philosophy of Science: Proceedings of the 1960 International Congress*. Stanford, CA: Stanford University Press, 1962, 240-251.
- [364] QUINE, WVO. Simple Theories of a Complex World. In *The Ways of Paradox*. New York: Random House, 1960, 242-246.
- [365] RACCOON, LBS. The Complexity Gap. *Software Engineering Notes*, 1995, 20, 37-44.
- [366] RAM, AL. A Study of Cognitive Complexity among College Students. *Journal of Psychological Researches*, 28, 84-90
- [367] RAMAMOORTHY, CV. An analysis of graphs by connectivity considerations. *Journal of the Association of Computing Machinery*, 1966, 13, 211-222.
- [368] RAMIREZ, DE. On 2 Nonequivalent Measures of Complexity. *IEEE Transactions on Systems, Man and Cybernetics*, 1989, 19, 1073-1077.
- [369] RAMSEY, FP. *The foundations of mathematics*. London: Routledge & Kegan Paul, 1950. [original 1929]
- [370] RAY, TS. Evolution and Complexity. In COWAN, G; PINES, D; MELTZER, D (Eds). *Complexity: Metaphors, Models and Reality*. Reading, MA: Addison-Wesley, 1994, 161-173.
- [371] RAY, TS. Evolution, Complexity, Entropy and Artificial Reality. *Physica D*, 1994, 75, 239-263.
- [372] REDHEAD, M. 1980 Models in Physics. *British Journal for the Philosophy of Science*, 31:145-163.
- [373] REGGIANI, A; NIJKAMP, P. Competition and complexity in spatially connected networks. *System dynamics review*, 1995, 11, 51-66.
- [374] REINBERGER, H-J. Experimental Complexity in Biology: Some Epistemological and Historical Remarks. *Philosophy of Science*, 1997, 64(proc), S245-S254.
- [375] DE REYCK, B; HERROELEN, W. On the use of the complexity index as a measure of complexity in activity networks. *European Journal of Operational Research*, 1996, 91, 347-366.
- [376] RICHMOND, SA. A Simplification of the Theory of Simplicity. *Synthese*, 1996, 107, 373-393.
- [377] RISSANEN, J. Modelling by the shortest data description. *Automatica*, 1978, 14, 465-471.
- [378] RISSANEN, J. Stochastic Complexity and the MDL Principle. *Econometric Reviews*, 1987, 6, 85-102.
- [379] RISSANEN, J. Complexity of Models. In ZUREK, WH (Ed). *Complexity, Entropy and the Physics of Information*. Redwood City, California: Addison-Wesley, 1990, 117-125.

- [380] ROBINSON, TT. Independence of Two Nice Sets of Axioms for the Propositional Calculus. *Journal for Symbolic Logic*, 1968, 33, 265-270.
- [381] ROBERTS, FS. *Measurement Theory: with applications to decision making, utility and the social sciences*. Reading, MA: Addison-Wesley, 1979.
- [382] ROCK, RA. Integrative Complexity and non-specific set as determinants of performance in a complex solving task. *Journal of Behavioural Science*, 1968, 1, 17-25.
- [383] ROSEN, BK. Syntactic Complexity. *Information and Control*, 1974, 24, 305-335.
- [384] ROSEN, R. Complexity and system descriptions. In HARNETT, WE (Ed). *Systems: Approaches, theories, Applications*. Dordrecht, Holland: D.Reidel, 1977, 169-175.
- [385] ROSEN, R. *Anticipatory Systems*. New York: Pergamon, 1985.
- [386] ROSEN, R. On Information and Complexity. In CASTI, JL; KARLQUIST, A (Eds). *Complexity, Language and Life: Mathematical Approaches*. Berlin: Springer, 1985, 174-195.
- [387] ROSEN, R. Organisms as Causal Systems Which Are Not Mechanisms: An Essay into the Nature of Complexity. In ROSEN, R (Ed). *Theoretical Biology and Complexity: Three Essays on the Natural Philosophy of Complex Systems*. London: Academic Press, 1985, 165-203.
- [388] ROSEN, R. *Life Itself - A Comprehensive Enquiry into the Nature, Origin and Fabrication of Life*. New York: Columbia University Press, 1991.
- [389] ROSEN, R. Bionics Revisited. In HAKEN, H; KARLQUIST, A; SVEDIN, U (Eds). *The Machine as Metaphor and Tool*. Berlin: Springer-Verlag, 1993, 87-100.
- [390] ROSENKRANTZ, RD. *Inference, Method and Decision*. Boston: Reidel, 1976.
- [391] ROTMAN, B. *Taking God out of mathematics and putting the body back*. Stanford, CA: Stanford University Press, 1993
- [392] ROUSE, WB; ROUSE, SH. Measures of Complexity of Fault Diagnosis Tasks. *IEEE Transactions on Systems, Man and Cybernetics*, 1979, 9, 720-727.
- [393] RUSSELL, S; SUBRAMANIAN, D. Mutual constraints on Representation and Inference. In *Machine Learning, Meta-Reasoning and Logics*, BRAZDIL, PB; KONOLIGE, K. (eds.), Boston, MA: Kluwer Academic, 1990, 85-106.
- [394] RUTHEN, R. Adapting to Complexity. *Scientific American*, 1993, 268, 130-135.
- [395] SAHAL, D. Elements of an Emerging theory of Complexity Per Se. *Cybernetica*, 1976, 19, 5-38.
- [396] SAHAL, D. System Complexity: Its Conceptions and Measurement in the Design of Engineering Systems. *IEEE Transactions on Systems, Man and Cybernetics*, 1976, 6, 440-445.
- [397] SAMBROOK, T; WHITEN, A. On the Nature of Complexity in Cognitive and Behavioural Science. *Theory and Psychology*, 1977, 7, 191-213
- [398] SANDERSON, WC. Predictability, Complexity and Catastrophe in a Collapsible Model of Population, Development, and Environmental Interactions. *International Institute for Applied Systems Analysis*, 1994.
- [399] SAUNDERS, PT; HO, MW. On the Increase in Complexity in Evolution. *Journal of Theoretical Biology*, 1981, 90, 515-530.
- [400] SAVAGE, JE. Complexity of Computing, 1976.
- [401] SCHAFFER, C. Overfitting Avoidance as Bias. *Machine Learning*, 1993, 10, 153-178.
- [402] SCHAFFER, EF. *Discovery and Explanation in Biology and Medicine*. Chicago: Chicago University Press, 1993.

- [403] SCHRUBEN, L; YCESAN, E. Complexity of simulation models: a graph theoretic approach. In EVANS, GW; MOLLAGHASEMI, M; RUSSELL, EC; et al. (Eds). *Proceedings of the 1993 Winter Simulation Conference*. Piscataway, NJ: IEEE, 1993, 641-649.
- [404] SEAMAN, JM; KOENIG, F. A Comparison Measures of Cognitive Complexity. *Sociometry*, 1974, 37, 375-390.
- [405] SEELEY, D; RONALD, S. The Emergence of Connectivity and Fractal Time in the Evolution of Random Digraphs. *Complex Systems* 92, Australia National University, 1992.
- [406] SELLARS, W. *Philosophical Perspectives*. Reseda, CA: Ridgeway Press, 1977.
- [407] SERRA, R. Some Remarks on Different Measures of Complexity for the Design of Self-organising Systems. In TRAPPL, R (Ed). *Cybernetics and Systems '88*. Dordrecht: Kluwer Academic, 1988, 141-148.
- [408] SHANNON, CE; WEAVER, W. *The Mathematical Theory of Communication*. Urbana, Illinois: University of Illinois Press, 1949.
- [409] SHUQIANG, Z. Cognitive Complexity and Written Production in English as a Second Language. *Language Learning*, 1987, 37, 469-481.
- [410] SIEGELMANN, HT. Computation Beyond the Turing Limit. *Science*, 1995, 268, 545-548.
- [411] SIMON, HA. The Organization of Complex Systems. In PATTEE, HH (Ed). *Hierarchy Theory - The challenge of Complex Systems*. New York: George Braziller, 1973, 1-27.
- [412] SIMON, HA. How Complex are Complex Systems? *Philosophy of Science Association*, 1976, 2, 507-522.
- [413] SIMON, HA. Complexity and the Representation of Patterned Sequences of Symbols. In SIMON, HA (Ed). *Models of Thought*. New Haven, Connecticut: Yale University Press, 1979, 292-306.
- [414] SIMON, HA. The Architecture of Complexity. In *The Sciences of the Artificial*. Cambridge, Massachusetts: MIT Press, 1981, 192-229.
- [415] SIMON, HA. *Economics, Bounded Rationality and the Cognitive Revolution*. Brookfield, Vermont: Edward Elgar, 1992.
- [416] SMITH, SW et al. Cognitive Complexity and Gifted Education, 1983.
- [417] SNEED, J. *The Logical Structure of Mathematical Physics*. Dordrecht: Reidel, 1971.
- [418] SOBER, E. *Simplicity*. Oxford: Clarendon Press, 1975.
- [419] SOLOMONOFF, RJ. A Formal theory of Inductive Inference. *Information and Control*, 1964, 7, 1-22,224-54.
- [420] SOLOMONOFF, RJ. Complexity-based induction systems: comparisons and convergence theorems. *IEEE Transactions on Information Theory*, 1978, 24, 422-432.
- [421] STEEL, M. The Complexity of Reconstructing Trees from Qualitative Characters and Subtrees. *Journal of Classification*, 1992, 1992, 91-116.
- [422] STEIN, D. *Lectures in the Sciences of Complexity*. Redwood City, California: Addison-Wesley, 1989.
- [423] STEVENS, SS. On the Theory of Scales of Measurement. *Science*, 1946, 103, 677-680.
- [424] STRASSEN, V. Gaussian elimination is not optimal. *Numerische Mathematik*, 1969, 13, 354-356.
- [425] SUBRAMANIAN, A. The Parallel Complexity of Propagation in Boolean Circuits. *Information and computation*, 1995, 117, 266.

- [426] SUEDFIELD, P; TETLOCK, PE; STREUFERT, S. Conceptual/Integrative Complexity. In SMITH, PS; ATKINSON, JW; MCCLELLAND, DC; et al. (Eds). *Motivation and Personality: Handbook of Thematic Content Analysis*. New York: CUP, 1992, 726.
- [427] SUPPE, F. The Search for Philosophical Understanding of Scientific Theories. Suppe, F (ed.). *The Structure of Scientific Theories*. Urbana, IL: University of Urbana Press, 1973, 3-243.
- [428] SUPPE, F. Exemplars, Theories and Disciplinary Matrixes. SUPPE, F (ed.). *The Structure of Scientific Theories*. Urbana, IL: University of Urbana Press, 1973, 483-499.
- [429] SUPPE, F. *The Semantic Conception of Theories and Scientific Realism*. Urbana, IL: University of Urbana Press, 1989.
- [430] SUPPES, P. *Introduction to Logic*. New York: van Nostrand, 1957.
- [431] SUPPES, P. Models of Data. In NAGEL, E; SUPPES, P; TARSKI, A (eds.). *Logic Methodology and the Philosophy of Science: Proceedings of the 1960 International Congress*. Stanford, CA: Stanford University Press, 1962, 252-261.
- [432] SUPPES, P. What is a Scientific Theory?. In MORGENBESSER, S. (Ed.). *Philosophy of Science Today*. New York: Basic Books, 1967, 55-67.
- [433] SUPPES, P. *Set theoretic Structures in Science*. Stanford: Stanford University Press, 1967.
- [434] SUPPES, P. Some Remarks about Complexity. *Philosophy of Science Association*, 1977, 2, 543-547.
- [435] TAYLOR, R. Causation. In *Encyclopedia of Philosophy*, P. Edwards, Macmillan, New York, 1967.
- [436] TEMPERLY, HNV. *Graph Theory and Applications*. Chichester: Ellis Horwood, 1981.
- [437] TIAN, J; ZELKOWITZ, V. Complexity Measure Evaluation and Selection. *IEEE Transactions on Software Engineering*, 1995, 21, 641-650.
- [438] TOULMIN, S. *The Philosophy of Science*. London: Hutchinson, 1953.
- [439] TRAPPL, R (Ed.). *Power, Autonomy, Utopia: New Approaches towards Complex Systems*. New York: Plenum Press, 1986.
- [440] TRAUB, JF; WASILKOWSKI, GW; WOZNIAKOWSKI, H. *Information, Uncertainty, Complexity*. Reading, MA: Addison-Wesley, 1983.
- [441] TRAUB, JF; WASILKOWSKI, GW; WOZNIAKOWSKI, H. *Information-Based Complexity*. London: Academic Press, 1988.
- [442] TRAUB, JF; WOZNIAKOWSKI, H. Information-Based Complexity: New Questions for Mathematicians. *Mathematical Intelligencer*, 1991, 13, 34-43.
- [443] TURING, AM. On computable Numbers with an Application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society*, 1936, 42, 230-265.
- [444] TURNEY, P. The Architecture of Complexity: A new blueprint. *Synthese*, 1989, 79, 515-542.
- [445] TURNEY, P. Problems with Complexity in Gold's Paradigm of Induction-Part I: Dynamic Complexity. *International Journal of General Systems*, 1990, 17, 329-342.
- [446] TURNEY, P. Problems with Complexity in Gold's Paradigm of Induction--Part II: Static Complexity. *International Journal of General Systems*, 1990, 17, 343-358.
- [447] TURNEY, P. The Curve Fitting Problem: A Solution. *British Journal for the Philosophy of Science*. 1990, 41, 509-530.
- [448] ULRICH, D. The Shortest Possible Length of the Longest Implicational Axiom. *Journal of Philosophical Logic*, 1996, 25, 101-108.

- [449] USPENSKY, VA. Kolmogorov and mathematical Logic. *Journal for Symbolic Logic*, 1992, 57, 385-411.
- [450] VAN EMDEN, MH. An Analysis of Complexity. Mathematical Centre Tracts 35, Mathematisch Centrum Amsterdam, Amsterdam, 1971
- [451] VAN LAMBAGEN, M. Algorithmic Information Theory. *Journal for Symbolic Logic*, 1989, 54, 1389-1400.
- [452] DE VANY, A. Information, Bounded Rationality, and The Complexity of Economic Organisation. Discussion paper, Dept. of mathematics, University. of California, 1993.
- [453] VITUSHKIN, AG. *Theory of Transmission and Processing of Information*. New York: Pergamon Press, 1961.
- [454] VON NEUMANN, J. *Theory of Self-Reproducing Automata*. Urbana, Illinois: University of Illinois Press, 1966.
- [455] VRANCKX, A. Narratives of “evolution”, “systems” and “complexity” - Semantic interpretations of realities and representation. In AERTS, D (ed.). *University and Society - Social and Cultural Studies of the Sciences*. Kluwer, Amsterdam, 1996.
- [456] VYUGIN, VV. On the Defect of Randomness of a Finite Object with respect to Measures with given Complexity Bounds. *Theory of Probability and its Applications*, 1988, 32, 508-512.
- [457] WACKERBAUER, R; WITT, A; ALTMANSPRACHER, H; et al. A Comparative Classification of Complexity Measures based on distinguishing partitions in phase space as well as structural v. dynamic elements. *Chaos, Solitons and Fractals*, 1994, 4, 133-173.
- [458] WALKER, CC. Behaviour of a Class of complex systems: The effect of systems size on properties of terminal cycles. *Journal of Cybernetics*, 1971, 1, 55-67.
- [459] WARFIELD, JN. *Planning, Policy, and Complexity*. New York: Wiley, 1976.
- [460] WARFIELD, JN. An Essay On Complexity. Centre for Advanced Study in the Integrative Sciences Report, Institute of Public Policy, George Mason University, Virginia, US, 1995.
- [461] WARFIELD, JN. Laws of Complexity. Center for Advanced Study in the Integrative Sciences Report, Institute of Public Policy, George Mason University, Virginia, US, 1996.
- [462] WARTOFSKY, MW. *Models*. Boston Studies in the Philosophy of Science, Vol. 129. Dordrecht: Reidel, 1979.
- [463] WAXMAN, M. Problem Complexity. Available electronically at URL: <http://www.fmb.mmu.ac.uk/~bruce/waxman.txt>, 1996.
- [464] WEAVER, W. Science and Complexity. *American Scientist*, 1948, 36, 536-544.
- [465] WEISS, L. *Statistical Decision Theory*. New York: McGraw-Hill, 1961.
- [466] WEYUKER, EJ. The Evaluation of Software Complexity Measures. *IEEE Transactions on Software Engineering*, 1988, 14, 1357-1365.
- [467] WHITEHEAD, AN; RUSSELL, B. *Principia Mathematica*. Cambridge: Cambridge University Press, 1910.
- [468] WIMSATT, W. Complexity and Organisation. In SCHAFFNER, K; COHEN, R (Eds). *Studies in the Philosophy of Sciences*. Dordrecht: Reidel, 1972, 67-86.
- [469] WINNIE, JA. Computable Chaos. *Philosophy of Science*, 1992, 59, 263.
- [470] WINOGRAD, S. Redundancy and Complexity of Logical Elements. *Information and Control*, 1963, 5, 177-194.
- [471] WOLFRAM, S. Cellular Automata as Models of Complexity. *Nature*, 1984, 311, 419-424.

- [472] WOLFRAM, S. Computational Theory of Cellular Automata. *Communications in Mathematical Physics*, 1984, 96, 15-57.
- [473] WOLFRAM, S. Universality and Complexity in Cellular Automata. *Physica D*, 1984, 10, 1.
- [474] WOLFRAM, S. Origins of Randomness in Physical Systems. *Physics Review Letters*, 1985, 55, 449-452.
- [475] WOLFRAM, S (Ed.). *Theory and Applications of Cellular Automata*. Singapore: World Scientific, 1986.
- [476] WOLFRAM, S. Complex Systems Theory. In PINES, D (Ed). *Emerging Syntheses in Science*. Redwood City, California: Addison-Wesley, 1987, 183-190.
- [477] WOOD, RE. Task Complexity: definition of the construct. *Organisational Behaviour and Human Decision Processes*, 1986, 36, 60-82.
- [478] WOODWARD, D. Understanding Complexity: A Critique and Synthesis. Henley-on-thames: Henley Management College, 1992.
- [479] WOODS, J. Relevance. *Logique et Analyse*, 1996, 27, 130-137.
- [480] XIE, H. Distinct Excluded Blocks and Grammatical Complexity of Dynamical Systems. *Complex Systems*, 1995, 9, 73-90.
- [481] YATES, FE. Complexity and the limits to knowledge. *American Journal of Physiology*, 1978, 235, R201-R204.
- [482] YATES, FE. Order and Complexity in Dynamical Systems - Homeodynamics as a Generalised Mechanics for Biology. *Mathematical and Computer Modelling*, 1994, 19, 49-74.
- [483] YNGVE, VH. A model and an hypothesis for language structure. *Proceedings of the American Philosophical Society*, 1960, 104, 444-466.
- [484] YOKOYAMA, T. Complexity and Hierarchy Theory. *Osaka Economic Papers*, 1976, 26, 26-34.
- [485] YOUNG, P; PARKINSON, S; LEES, M. Simplicity out of complexity in environmental modelling: Occam's razor revisited. *Journal of Applied Statistics*, 1996, 23, 165-210.
- [486] YUDIN, D; GORYASHKO, A. Control problems and Complexity Theory I. *Engineering Cybernetics*, 1974, 12, 10-24.
- [487] YUDIN, D; GORYASHKO, A. Control problems and Complexity Theory II. *Engineering Cybernetics*, 1975, 13, 1-13.
- [488] ZADEH, LA. Outline of a new approach to the analysis of complex systems and decision processes. *IEEE Transactions on Systems, Man and Cybernetics*, 1973, 3, 28-44.
- [489] ZADEH, LA. Coping with the imprecision of the real world. In YAGER, RR et al. (eds.). *Selected Papers by Lofti A. Zadeh*. New York: John Wiley, 1987, 9-28.
- [490] ZAMBELLA, D; GRASSBERGER, P. Complexity of Forecasting in a Class of Simple Models. *Complex Systems*, 1988, 2, 269-303.
- [491] ZEIGARNIK, AV; TEMKIN, ON. A graph-theoretic model of complex reaction mechanisms: a new complexity index for reaction mechanisms. *Kinetics and Catalysis*, 1996, 37, 372-385.
- [492] ZEIGLER, BP. Simulation based structural complexity of models. *International Journal of General Systems*, 1976, 2, 217-223.
- [493] ZHANG, B-T; MÜHLENBEIN, H. Balancing accuracy and parsimony in genetic programming. *Evolutionary Computation*, 1995, 3, 17-38.

- [494] ZIMAND, M. A High-Low Kolmogorov Complexity Law equivalent to the 0-1 Law. *Information Processing Letters*, 1996, 57, 59-64.
- [495] ZIV, J. Complexity and Coherence of Sequences. In DURAND, H; DI LULLO, M; SINCLAIR, C (Eds). *The Impact of Data Processing Techniques on Communications*. Dordrecht: Nijhoff, 1983, 23-33.
- [496] ZUREK, WH. Algorithmic Randomness and Physical Entropy. *Physics Review A*, 1989, 40, 4731-4751.
- [497] ZUREK, WH. Algorithmic Information Content, Church-Turing Thesis, Physical Entropy, and Maxwell's Demon. In ZUREK, WH (Ed). *Complexity, Entropy and the Physics of Information*. Redwood City, California: Addison-Wesley, 1990, 73-89.
- [498] ZUSE, H. *Software Complexity*. Berlin: Walter de Gruyter, 1991.
- [499] ZVONKIN, AK; LEVIN, LA. The Complexity of Finite Objects and the Development of the Concepts of Information and Randomness by Means of the Theory of Algorithms. *Russian Mathematics Surveys*, 1970, 256, 83-124.
- [500] ZWIETERING, PJ; AARTS, EHL; WESSELS, J. Design and Complexity of Exact Multi-Layered Perceptrons. *International Journal of Neural Systems*, 1994, 2, 185-201.