Titles in the series

1. G. Chaitin *Algorithmic Information Theory*
2. L. C. Paulson *Logic and Computation*
3. M. Spivey *Understanding Z*
4. G. Revesz *Lambda Calculus, Combinators and Functional Programming*
5. A. Ramsay *Formal Methods in Artificial Intelligence*
6. S. Vickers *Topology via Logic*
7. J.-Y. Girard, Y. Lafont & P. Taylor *Proofs and Types*
8. J. Clifford *Formal Semantics & Pragmatics for Natural Language Processing*
9. M. Winslett *Updating Logical Database*
10. K. McEvoy & J. V. Tucker (eds) *Theoretical Foundations of VLSI Design*
11. T. H. Tse *A Unifying Framework for Stuctured Analysis and Design Models*
12. G. Brewka *Nonmonotonic Reasoning*
14. S. G. Hoggar *Mathematics for Computer Graphics*
15. S. Dasgupta *Design Theory and Computer Science*
17. J. C. M. Baeten (ed) *Applications of Process Algebra*
18. J. C. M. Baeten & W. P. Weijland *Process Algebra*
21. D. A. Wolfram *The Clausal Theory of Types*
23. E.-R. Olderog *Nets, Terms and Formulas*
26. P. D. Mosses *Action Semantics*
27. W. H. Hesselink *Programs, Recursion and Unbounded Choice*
28. P. Padawitz *Deductive and Declarative Programming*
29. P. Gärdenfors (ed) *Belief Revision*
30. M. Anthony & N. Biggs *Computational Learning Theory*
31. T. F. Melham Higher *Order Logic and Hardware Verification*
32. R. L. Carpenter *The Logic of Typed Feature Structures*
33. E. G. Manes *Predicate Transformer Semantics*
34. F. Nielson & H. R Nielson *Two Level Functional Languages*
35. L. Feijs & H. Jonkers *Formal Specification and Design*
36. S. Mauw & G. J. Veltink (eds) *Algebraic Specification of Communication Protocols*
37. V. Stavridou *Formal Methods in Circuit Design*
38. N. Shankar *Metamathematics, Machines and Gödel's Proof*
39. J. B. Pairs *The Uncertain Reasoner's Companion*
40. J. Desel & J. Esparza *Free Choice Petri Nets*
41. J.-J. Ch. Meyer & W. van der Hoek *Epistemic Logic for AI and Computer Science*

INFORMATION FLOW

The Logic of Distributed Systems

Jon Barwise
Indiana University

Jerry Seligman
National Chung-Cheng University, Taiwan

CAMBRIDGE
UNIVERSITY PRESS

PUBLISHED BY THE PRESS SYNDICATE OF THE UNIVERSITY OF CAMBRIDGE
The Pitt Building, Trumpington Street, Cambridge CB2 1RP, United Kingdom

CAMBRIDGE UNIVERSITY PRESS
The Edinburgh Building, Cambridge CB2 2RU, United Kingdom
40 West 20th Street, New York, NY 10011-4211, USA
10 Stamford Road, Oakleigh, Melbourne 3166, Australia

First published 1997

Printed in the United States of America

Typeset in Times Roman

Library of Congress Cataloging-in-Publication Data

Barwise, Jon.
Information flow: the logic of distributed systems / Jon Barwise,
Jerry Seligman.
p. cm. – (Cambridge tracts in theoretical computer science:
44)
Includes bibliographical references and index.
ISBN 0-521-58386-1
1. Electronic data processing – Distributed processing.
I. Seligman, Jarry, 1964– , II. Title. III. Series.
QA76.9.D5B363 1997
003′.54 – dc20 96-46111
CIP

A catalog record for this book is available from
the British Library

ISBN 0 521 58386 1 hardback

In the same way, the world is not the sum of all the things that are in it. It is the infinitely complex network of connections among them. As in the meanings of words, things take on meaning only in relationship to each other.

The Invention of Solitude Paul Auster

Contents

Preface

Information and talk of information is everywhere nowadays. Computers are thought of as information technology. Each living thing has a structure determined by information encoded in its DNA. Governments and companies spend vast fortunes to acquire information. People go to prison for the illicit use of information. In spite of all this, there is no accepted science of information. What *is* information? How is it possible for one thing to carry information about another? This book proposes answers to these questions.

But why does information matter, why is it so important? An obvious answer motivates the direction our theory takes. Living creatures rely on the regularity of their environment for almost everything they do. Successful perception, locomotion, reasoning, and planning all depend on the existence of a stable relationship between the agents and the world around them, near and far. The importance of regularity underlies the view of agents as information processors. The ability to gather information about parts of the world, often remote in time and space, and to use that information to plan and act successfully, depends on the existence of regularities. If the world were a completely chaotic, unpredictable affair, there would be no information to process.

Still, the place of information in the natural world of biological and physical systems is far from clear. A major problem is the lack of a general theory of regularity. In determining which aspects of the behavior of a system are regular, we typically defer to the scientific discipline suited to the task. For regularities in the movements of birds across large distances, we consult experts in ornithology; but if the movements are of bright lights in the night sky, we'd be better off with an astronomer. Each specialist provides an explanation using a theory or model suited to the specialty.

To whom can we turn for questions about information itself? Can there be a science of information? We think so and propose to lay its foundations in this book. Out of the wide variety of models, theories, and less formal modes of explanation used by specialists, we aim to extract what is essential to understanding the flow of information.

How to Read This Book

This book has several intended audiences, some interested in the big picture but not too concerned about the mathematical details, the other of the opposite sort. Here is a brief outline of the book, followed by indications of which parts various readers will want to read, and in what order.

Introduction to the Theory. The book has three parts. Part I contains a discussion of the motivations for a model of this kind and surveys some related work on information. An overview of our information-channel model is presented, and a simple but detailed example is worked out.

Development of the Theory. The heart of the book is in Part II and consists of a detailed elaboration of the mathematical model described in Part I. Although the basic picture presented in Part I is reasonably simple, there is a lot of mathematical spadework to be done to fill in the details. The mathematical part of the book culminates in a theory of inference and error using "local logics" in Lectures 12–16.

Applications of the Theory. The lectures in Part III explore some ideas for an assortment of applications, namely, applications to speech acts, vagueness, commonsense reasoning (focusing on monoticity and on the frame problem), representation, and quantum logic. Although there is some truth to the old saw that to a person with a hammer, everything looks like a nail, a wise person with a new tool tests it to see what its strengths and weaknesses are. It is in this spirit that we offer the "explorations" of Part III. We are interested in pounding nails but are equally interested in exploring our new hammer.

We have written the book so that philosophers and others less patient with mathematical detail can read Part I and then have a look through Part III for topics of interest. Mathematicians, logicians, and computer scientists less patient with philosophical issues might prefer to start with Lecture 2, and then turn to Part II, followed perhaps by poking around in Part III. Researchers in artificial intelligence would probably also want to start with Lecture 2 and some of the chapters in Part III, followed by a perusal of Parts I and II.

An index of definitions used in Part II, and a glossary of special notation used in Part II, can be found at the end of the book.

Mathematical Prerequisites. Although some familiarity with modern logic would be helpful in understanding the motivations for some of the topics we cover, the book really only requires familiarity with basic set theory of the sort used in most undergraduate mathematics courses. We review here some more or less standard notational conventions we follow.

If $a = \langle x, y \rangle$ is an ordered pair, then we write $1^{st}(a) = x$ and $2^{nd}(a) = y$. Given a function $f: X \to Y$, we write $f[X_0] = \{f(x) \mid x \in X_0\}$ (for $X_0 \subseteq X$) and $f^{-1}[Y_0] = \{x \in X \mid f(x) \in Y_0\}$ (for $Y_0 \subseteq Y$). By the range of f we mean $f[X]$. Given functions $f: X \to Y$ and $g: Y \to Z$, gf is the function that results from composing them to get the function $gf: X \to Z$ defined by $gf(x) = g(f(x))$. For any set A, pow A is its power set, that is, the set of all its subsets. If $\mathcal{X}$ is a set of subsets of A, then

$$\bigcup \mathcal{X} = \{x \in A \mid x \in X \text{ for some } X \in \mathcal{X}\}$$

and

$$\bigcap \mathcal{X} = \{x \in A \mid x \in X \text{ for all } X \in \mathcal{X}\}.$$

If $\mathcal{X}$ is empty, then so is $\bigcup \mathcal{X}$ but $\bigcap \mathcal{X} = A$.

The appearance of many diagrams may suggest that the book uses category theory. This suggestion is both correct and misleading. Category theory arose from the realization that the same kinds of diagrams appear in many branches of mathematics, so it is not surprising that some of these diagrams appear here. We must confess that we have found the basic perspective of category theory to be quite helpful as a guide in developing the theory. And, as it turned out, some of the category-theoretic notions (coproducts and, more generally, colimits) have an important information-theoretic interpretation. In writing this book we have tried to make clear the debt we owe to ideas from category theory, but, at the same time, not presuppose any familiarity with category theory, except in those exercises marked by a (†).

In mathematics it is typical practice to call the hard results "theorems" and the easier results "propositions." Because none of our results are very hard, we use "theorem" to designate the results that are most important to the overall theory we are developing. As a result, some of our theorems are simpler than some of our propositions.

World Wide Web Site. We have set up a home page for this book, to facilitate the distribution of developments based on the ideas presented here, as well as for any corrections that may need to be made. Its URL is

`http : //www.phil.indiana.edu/~barwise/ifpage.html.`

You can also send us e-mail from this page.

Acknowledgments

This book has been in the making for well over a decade. Barwise's attempts at trying to understand information flow date back at least to his book written with John Perry, *Situations and Attitudes*, which appeared in 1983, and include the following papers: Barwise (1986), Barwise (1983), Barwise (1989), Barwise (1991), Barwise and Etchemendy (1990), Barwise, Gabbay, and Hartonas (1996), Barwise (1993), and Barwise and Seligman (1994). Seligman has been working on the same topic since starting his dissertation, which was completed in 1991; see Seligman (1990), Seligman (1991b), Seligman (1991a), and Barwise and Seligman (1994). The theory developed in this book is the product of an intense collaboration between the authors over the past four years. We started working together on the topic during 1992–93, when Seligman was at Indiana University on a postdoctoral fellowship. Two drafts of Parts I and II were written during the academic year 1994–95, when Seligman returned to Indiana University for a year's visit. The explorations in Part III were undertaken by Barwise in 1995–96. During that year, Parts I and II were revised by the two of us, exploiting regularities of the internet to facilitate massive amounts of information flow.

Both authors have previously contributed to the development of situation theory, a programmatic theory of information whose roots go back to *Situations and Attitudes*, discussed at length in Barwise (1989) and further developed in Devlin (1991). Though not presupposing familiarity with situation theory, the present book can be seen as a contribution to that subject by considering those tokens that are situations or events, classified using "infons" as types. Indeed, we consider the notion of a constraint to be the most important notion of situation theory and have tried, in this book, to make this notion less problematic. The task of reconciling the development of the ideas presented here with other details of situation theory remains to be undertaken.

We express our gratitude to Robin Cooper for providing the initial connection between the two of us, to Faber and Faber for permission to use the quotation from Paul Auster's *The Invention of Solitude*, to Indiana University (I.U.) for support of many kinds, to the I.U. Computer Science and Philosophy class

that worked through an earlier and much more opaque version during the fall of 1994, to the Situation Theory and Situation Semantics seminar at the Center for the Study of Language and Information (CSLI) for their interest in these topics over the years, to Imperial College for hospitality and support while Barwise wrote a draft of the first two lectures during July of 1995, to the members of the I.U. Program in Pure and Applied Logic for their support over the years, to Michael Dickson and Kenneth Kunen for helpful comments on the quantum logic chapter, to various readers, including Gerry Allwein, Kata Bimbo, Tony Chemero, Diarmuid Crowley, Steve Crowley, Albert Chapman-Leyland, Keith Devlin, Yuko Murakami, and an anonymous referee for helpful comments that have helped us improve the book, to Mary Jane Wilcox for assistance with proofreading, to Lauren Cowles of Cambridge University Press for her help of many kinds, to our families for putting up with us as we wrestled with the ideas that led to the theory, and to the members of the audiences at many talks we have given on this material over the past few years for their interest and questions. Finally, we thank you, the reader, for your attention to our book.

August 1996

JON BARWISE
JERRY SELIGMAN

Part I

Introduction

Lecture 1

Information Flow: A Review

The three lectures in the first part of the book present an informal overview of a theory whose technical details will be developed and applied later in the book. In this lecture we draw the reader's attention to the problems that motivated the development of the theory. In Lecture 2 we outline our proposed solution. A detailed example is worked out in Lecture 3.

In the course of the first two lectures we draw attention to four principles of information flow. These are the cornerstones of our theory. We do not attempt to present a philosophical argument for them. Rather, we illustrate and provide circumstantial evidence for the principles and then proceed to erect a theory on them. When the theory is erected, we shall be in a better position to judge them. Until that job is done, we present the principles as a means of understanding the mathematical model to be presented – not as an analysis of all and sundry present day intuitions about information and information flow.

1.1 The Worldly Commerce of Information

In recent years, information has become all the rage. The utopian vision of an information society has moved from the pages of science fiction novels to political manifestos. As the millennium approaches fast on the information highway, the ever-increasing speed and scope of communication networks are predicted to bring sweeping changes in the structure of the global economy. Individuals and companies are discovering that many transactions that used to require the movement of people and goods, often at great expense, may now be accomplished by the click of a mouse. Information can travel at the speed of light; people and goods cannot.[1] The result is no less than a reshaping of

[1] If information about the future is possible, as it seems to be, then information travels *faster* than the speed of light.

our shrinking planet as cultural and commercial boundaries are transformed, for better or worse, by the increasing volume of information flow.

No doubt such future-mongering should be taken with more than a pinch of salt, but there can be little doubt that the prospect of life in "cyberspace" has caught the imagination of our age. Even in the most sober of society's circles, there is a mixture of heady excitement and alarm, a sense of revolution of an almost metaphysical sort.

Once one reflects on the idea of information flowing, it can be seen to flow everywhere – not just in computers and along telephone wires but in every human gesture and fluctuation of the natural world. Information flow is necessary for life. It guides every action, molds every thought, and sustains the many complex interactions that make up any natural system or social organization. Clouds carry information about forthcoming storms; a scent on the breeze carries information to the predator about the location of prey; the rings of a tree carry information about its age; a line outside the gas station carries information about measures in the national budget; images on a television screen in Taiwan can carry information about simultaneous events in Britain; the light from a star carries information about the chemical composition of gases on the other side of the universe; and the resigned shrug of a loved one may carry information about a mental state that could not be conveyed in words.

With this perspective, the current revolution appears to be primarily technological, with people discovering new and more efficient ways to transform and transmit information. Information is and always was all around us, saturating the universe; now there are new ways of mining the raw material, generating new products, and shipping them to increasingly hungry markets.

This book, however, is not concerned with technology. Our primary interest is not so much in the ways information is processed but in the very possibility of one thing carrying information about another. The metaphor of information flow is a slippery one, suggesting the movement of a substance when what occurs does not necessarily involve either motion or a substance. The value of the metaphor lies largely in the question it raises: How *do* remote objects, situations, and events carry information about one another without any substance moving between them?

The question is not a new one. A variety of answers have been proposed by philosophers, mathematicians, and computer scientists. Our starting point was the work of Dretske, which will be discussed below. However, before going into details it is worth asking what such an answer is meant to achieve.

Consider the following story:

Judith, a keen but inexperienced mountaineer, embarked on an ascent of Mt. Ateb. She took with her a compass, a flashlight, a topographic map, and a bar of Lindt bittersweet

chocolate. The map was made ten years previously, but she judged that the mountain would not have changed too much. Reaching the peak shortly after 2 P.M. she paused to eat two squares of chocolate and reflect on the majesty of her surroundings.

At 2:10 P.M. she set about the descent. Encouraged by the ease of the day's climb, she decided to take a different route down. It was clearly indicated on the map and clearly marked on the upper slopes, but as she descended the helpful little piles of stones left by previous hikers petered out. Before long she found herself struggling to make sense of compass bearings taken from ambiguously positioned rocky outcrops and the haphazard tree line below. By 4 P.M. Judith was hopelessly lost.

Scrambling down a scree slope, motivated only by the thought that down was a better bet than up, the loose stones betrayed her, and she tumbled a hundred feet before breaking her fall against a hardy uplands thorn. Clinging to the bush and wincing at the pain in her left leg, she took stock. It would soon be dark. Above her lay the treacherous scree, below her were perils as yet unknown. She ate the rest of the chocolate.

Suddenly, she remembered the flashlight. It was still working. She began to flash out into the twilight. By a miracle, her signal was seen by another day hiker, who was already near the foot of the mountain. Miranda quickly recognized the dots and dashes of the SOS and hurried on to her car where she phoned Mountain Rescue. Only twenty minutes later the searchlight from a helicopter scanned the precipitous east face of Mt. Ateb, illuminating the frightened Judith, still clinging to the thorn bush but now waving joyously at the aircraft.

Two previously unacquainted people, Judith and the helicopter pilot, met on the side of a mountain for the first time. How did this happen? What is the connection between the helicopter flight and Judith's fall, such that the one is guided by the location of the other?

Naturally, common sense provides answers – in broad outline, at least. We explain that the helicopter was flying over that part of the mountain because the pilot believed that there was someone in danger there. The location had been given to the pilot at the Mountain Rescue Center shortly after the telephone operator had turned Miranda's description into a range of coordinates. Miranda also conveyed the description by telephone, but her information was gained from the flashes of light coming from Judith's flashlight, which was responding to the desperate movements of Judith's right thumb as she clung to the mountainside.

This establishes a physical connection between the two events, but a lot is left unsaid. Most events are connected in one way or another. How is this particular connection capable of conveying the vital piece of information about Judith's location? Consider the nature of the connection. It is made up of a precarious thread of thoughts, actions, light, sound, and electricity. What is it about each of these parts and the way they are connected that allows the information to pass?

A full explanation would have to account for all the transitions. Some of them may be explained using existing scientific knowledge. The way in which the switch on Judith's flashlight controlled the flashes, the passage of light from the flashlight to Miranda's eyes, and the transformation of spoken words into electrical signals in telephone wires and radio waves may all be explained using models derived ultimately from our understanding of physics. The motion of Judith's thumb muscles and the firing of Miranda's retinal cells, as well as the many other critical processes making up the human actions in this story, require physiological explanation. A knowledge of the conventions of mapmakers and English speakers are needed to explain the link between the combinations of words, map coordinates, and the actual location on the mountain. And, finally, the psychology of the various people in the story must be understood to bridge the considerable gap between perception and action: between Judith's fall and her moving the switch, between the light falling on Miranda's retina and her mouthing sounds into a cellular phone, between the sounds coming from the telephone, the scribbled calculations on a message pad, and the pilot's hurried exit from the building.

A full explanation would have to include all these steps and more. It is no wonder that we speak of information, knowledge, and communication; life is too short to do without them. Yet it is not just the complexity of the explanation that makes the prospect of doing without information-based vocabulary so daunting. Stepping below the casual uniformity of talk about information, we see a great disunity of theoretical principles and modes of explanation. Psychology, physiology, physics, linguistics, and telephone engineering are very different disciplines. They use different mathematical models (if any), and it is far from clear how the separate models may be linked to account for the whole story. Moreover, at each stage, we must ask why the information that someone is in danger on the east face of Mt. Ateb is carried by the particular event being modeled. This question is not easily stated in terms appropriate to the various models. To explain why the pattern and frequency of firings in Miranda's retinas carry this information, for instance, we need more than a model of the inside of her eye; the question cannot even be stated in purely physiological terms.

What are the prospects for a rigorous understanding of the principles of information flow? It is relatively uncontroversial that the flow of information is ultimately determined by events in the natural world and that the best way of understanding those events is by means of the sciences. But explanations based on the transfer of information are not *obviously* reducible to scientific explanations, and even if they are, the hodgepodge of models and theoretical principles required would quickly obscure the regularities on which the information-based explanations depend. The possibility exists that a rigorous model of information flow can be given in its own terms; that such phenomena as the chaining together of information channels, the relationship between error and gaps in the chain, and the difference between a reliable information source and accidental correlations can be explained in a precise way, not by reducing the problem to physics or any other science, but by appealing to laws of information flow. This book aims to provide such a model.

1.2 Regularity in Distributed Systems

In determining the information carried by Judith's flashlight signal, a spectrograph is of little use. Whatever information is carried depends not on intrinsic properties of the light but on its relationship to Judith and other objects and events.

It is all too easy to forget this obvious point if one focuses on information conveyed by means of spoken or written language. The information in a newspaper article appears to depend little on the particular copy of the newspaper one reads or on the complex mechanism by which the article was researched, written, printed, and distributed. The words, we say, speak for themselves. The naïviety of this remark is quickly dispelled by a glance at a newspaper written in an unfamiliar language, such as Chinese. There is nothing in the intricate form of Chinese characters inscribed on a page that conveys anything to the foreigner illiterate in Chinese.

Logical and linguistic investigations into the topic of information give the impression that one should be concerned with properties of sentences. Even when it is acknowledged that information is not a syntactic property of sentences and that some system of interpretation is required to determine the information content of a sentence, the role of this system is typically kept firmly in the background. In Tarskian model theory, for example, and in the approaches to natural language semantics based on it, an interpretation of a language consists of an abstract relation between words and sets of entities. No attempt is made to model what it is about human language-using communities that makes this relation hold.[2]

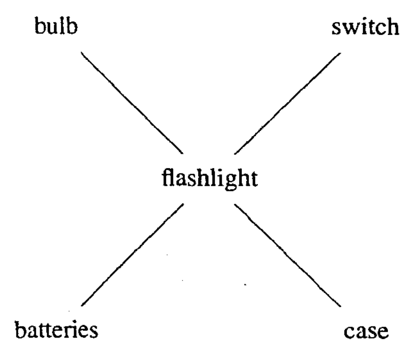
By contrast, when one looks at nonlinguistic forms of communication, and the many other phenomena that we have listed as examples of information flow, the spatial and temporal relationships between parts of the system cannot be ignored. The very term "information flow" presupposes, albeit metaphorically, a spatial and temporal separation between the source and the receiver of the information.

[2] The criticism is not that the topic has been ignored, as philosophers of language have had much to say about it, but that their proposals have not been incorporated into logico-linguistic theories. The relation between a name and its bearer, for example, is taken to be a primitive relation of semantic theory; the contingent facts of language use that establish the relation are ignored on the grounds that the only semantically relevant feature of a name is its bearer.

We draw attention to the importance of examining information flow in the context of a system in our first Principle.

The First Principle of Information Flow: Information flow results from regularities in a distributed system.

Two features of this principle require immediate comment: that the systems in which information flows are *distributed* and that the flow results from *regularities* in the system. By describing a system in which information flows as "distributed," we mean that there is some way in which it is divided into parts, so that the flow of information is from one part (or parts) to another. For example, we may consider Judith's flashlight to be a system in which information flows: the lit bulb carries the information that the switch is on and the battery is charged and so forth. The flashlight may be divided into the following parts:



We do not intend to suggest that this division into parts is unique or comprehensive. Each of the components of the flashlight has parts that are not represented (but could be) and there are a host of different ways of decomposing the system that would not list any of the parts depicted above. (Indeed, the relativity of this decomposition is crucial to the story we will tell.)

The parts of an information system are often spatial or temporal parts but they do not need to be. The conception of information flow we develop is very broad, encompassing abstract systems such as mathematical proofs and taxonomic hierarchies as well as concrete ones like the above. In abstract systems the relation of whole to part is also abstract, and the metaphor of "flow" has to be interpreted even more loosely. We place no restriction on what kind of thing may count as a part, only that the choice of parts determines the way in which we understand what it is for information to flow from one part to another.

The first principle of information flow also states that information flow results from *regularities in the system*. It is the presence of regularities that links the parts of a system together in a way that permits the flow of information. The fact that the components of Judith's flashlight are more or less adjacent in space and have overlapping temporal extents is not sufficient to explain the information carried by the lit bulb about the position of the switch or the state of the batteries. It is of the greatest importance that the components of the flashlight are connected together in such a way that the whole flashlight behaves in a more or less predictable manner. In this case, the regularities that ensure the uniform behavior of the system are mostly electrical and mechanical in nature. The contacts in the switch, the design of the case, and many other details of the construction of the flashlight go to ensure that the flashing of the bulb is systematically related to the position of the switch.

The behavior of the system need not be entirely predictable for information to flow. Properties of components of the flashlight, such as the discoloring of the plastic case due to exposure to sunlight, are not at all predictable from properties of the other components; yet this indeterminacy does not interfere with the regular behavior of the system in the informationally relevant respects. More complex systems may even be highly nondeterministic while still allowing information flow. Yet, as a general rule, the more random the system the less information will flow.

The range of examples in the story of Judith on Mt. Ateb show that information flows may be due to a wide range of factors. Some of them are "nomic" regularities, of the kind studied in the sciences; others, such as those relating a map to the mapped terrain, are conventional; and others are of a purely abstract or logical character. Sometimes the regularity involved is very difficult to pin down. Parents can often tell when their children are getting ill just from the look of their eyes. The relationship between the appearance of the eyes and the condition permits this inference reliably, but even an ophthalmologist would hesitate to say on what exactly the inference depends.

Despite the wide range of regularities that permit information flow, it is important to distinguish genuine regularities from merely accidental, or "statistical" regularities. Accidents are not sufficient for information flow. To give an example, suppose that on the occasion of Judith's fall, Miranda caught sight of the light of the moon reflected from a small waterfall on the mountainside. By chance, we suppose, the waterfall reflected the moonlight in a sequence of flashes very similar to the Morse code SOS. In such circumstances, Miranda might have formed the belief that there was someone in trouble on the mountain in the approximate location of Judith's fall. Her belief would be true but she would not have the information. Information does not flow from A to B just because someone at B happens to have been misled into believing something correct about what is going on at A.

Now suppose that over the course of several months, a large number of climbers are saved from the treacherous slopes of Mt. Ateb by Mountain Rescue, alerted by reports of SOS signals flashed from the mountain. In fact, the flashes were all caused by reflected moonlight and a spate of visiting climbers, rather overzealous in their desire to report potential accidents. It is important that we assume that there really is no more to it that this, that is, no mysterious link between troubled climbers and water spirits. The result would be that a statistical regularity is established between flashes from the mountainside and climbers in distress. It should be clear that this spurious regularity is no more able to establish information flow than the one-time coincidence considered previously.[3]

1.3 Information and Knowledge

There is a close connection between information and knowledge. Puzzles similar to those discussed in the previous section are used by philosophers to test different theories of knowledge. Indeed, the origin of the work presented here was an attempt to elaborate and improve on a theory of knowledge presented by Fred Dretske (1981) in his book *Knowledge and the Flow of Information*. The informational role of regularities in distributed systems will be better appreciated if seen in the light of Dretske's theory.

Since Gettier's famous paper of 1963, philosophers have been looking for the missing link between true belief and knowledge. The "traditional" account is that knowledge is *justified* true belief. Miranda knows that someone is in trouble on the mountain because her belief is justified by her knowledge of Morse code and other relevant considerations. But consider the case in which the flashes were produced by reflected moonlight. Miranda's belief and its justification would remain the same, but she would not *know* that someone is in trouble. This is Gettier's argument and there have been many responses to it.[4]

Recently, the topic has largely been put aside, not because an agreed upon solution has been found but because many have been proposed and no clear victor has emerged. Dretske's solution was one of the first. He proposed that information is the missing link. Very roughly, Dretske claims that a person knows that p if she believes that p and her believing that p (or the events in her head responsible for this belief) carries the information that p. To the extent

[3] An interesting question, which tests the best of intuitions, is whether information is carried by Judith's flashlight signals against a background of fortuitously correct accident reports of the kind considered in our example.

[4] For a survey, see Shope (1983).

that our beliefs carry information about the world, they play an invaluable role in guiding our actions and in communicating with others.

We take Dretske's account of the relationship between information and knowledge to be an important insight.[5] As a bridge between two subjects, we can use it both as a means of applying our theory to epistemology and also as a way of incorporating epistemological considerations into the theory of information.

For example, the first principle of information flow is illuminated by considering some related epistemological puzzles. A perennial problem in the philosophy of knowledge is that of accounting for a person's knowledge of remote facts. Miranda's knowledge of the world must stem from her experience of it, and yet a great deal of her knowledge concerns things that are not and never were part of our immediate physical environment. She may never have climbed Mt. Ateb herself, and she is certainly not able to see what happened on the scree slope in the dark. How is she able to know about parts of the world that are beyond her experience?

A rough answer is as follows: Things outside Miranda's experience are connected in lawlike ways to things within her experience. If the world beyond her senses bore no relationship to her experience, then she would not be able to know about it. It is the regularity of the relationship that makes knowledge possible. This answer places the philosophical problem squarely within the scope of our current investigations. Miranda and the remote objects of her knowledge form a distributed system governed by regularities. The fact that the system is distributed gives rise to the problem; the regularity of the system provides the only hope of a solution.

Dretske's approach to the theory of knowledge is not dissimilar to those who claim that it is the *reliability* of the belief-producing process that constitutes the difference between knowledge and mere true belief (Goldman, 1979, 1986; Nozick, 1981; Swain, 1981). Indeed, there is a close connection between information flow and reliability. For a signal to carry information about a remote state of affairs, it must have been produced by a reliable process. An unreliable process will not permit information to flow.

Reliability, however, is clearly a matter of degree. Some processes are more reliable than others, and what counts as sufficiently reliable may vary with circumstances. Consider Judith's flashlight. The information that the bulb is lit carries the information that the switch is on, because they are linked by a

[5] We do not accept Dretske's account of information based on probability theory, and motivation for the design of our theory can be traced back to inadequacies in his proposals. Dretske's proposals will be discussed in more detail in Section 1.5 of the present chapter.

reliable mechanism. But the mechanism can become less reliable. If the battery is dead, the switch could be on but the bulb not lit. If there is a short circuit, the switch might be off but the bulb lit. Considerations of this kind make one wonder whether the bulb being lit really carries the information that the switch is on even when the flashlight is "working properly." Much depends on the standards of reliability being used.

Consider also Judith's map of Mt. Ateb. Such a map is full of information, vital to the mountain navigator. But it also contains a number of inaccurate details – details that misled Judith into taking the wrong path and resulted in her perilous descent of the scree slope. The map is reliable in some respects but fallible in others; it carries both information and misinformation. Judith's use of the map is partly responsible both for her knowledge of the mountainous terrain around her and for her mistakes.

Whenever one talks of information, the issue of reliability is close at hand. A goal of this book is to present a theory of information that is compatible with the facts about reliability, especially the treatment of partially reliable information sources and the ubiquitous possibility of error.

1.4 The Grammar of Information Flow

There are no completely safe ways of talking about information. The metaphor of information flowing is often misleading when applied to specific items of information, even if the general picture is usefully evocative of movement in space and time. The metaphor of information content is even worse, suggesting as it does that the information is somehow intrinsically contained in the source and so is equally informative to everyone and in every context.

Perhaps the least dangerous is the metaphor of carrying information. One can at least make sense of one thing carrying different items of information on different occasions and for different people, and the suggestion of movement and exchange adds a little color. What's more, there is a pleasant variety of words that may be used as alternatives to "carry": "bear," "bring," and "convey" can all be used in the appropriate context.

We therefore adopt the form "x carries/bears/conveys the information that y" as our primary means of making claims about the conveyance of information. Some comment about the values of x and y is in order.

The position occupied by y may be filled by any expression used in the attribution of propositional attitudes: the values for y in "She knows/believes/doubts/thinks/hopes that y." Company is comforting, but in this case especially, the sense of security may well be false. The problem of determining the semantic role of such expressions is notoriously difficult and we do not wish to

suggest that anything we say about information involves a commitment to any alleged solution.

The position occupied by x may be filled by a variety of noun phrases:

The e-mail message bore the information that Albert would be late for dinner.
The rifle shot carried the information that the king was dead to the whole city.
Jane's being taller than Mary carries the information that Mary will fit under the arch.
The greyness of the sky carries the information that a storm is approaching.
The beating of drums conveyed the information that the enemy had been sighted.
Mary's kissing John carried the information that she had forgiven him.

It appears difficult to summarize this list. Events, such as Mary's kissing John, can carry information, so can objects, such as the e-mail message, and properties, such as the greyness of the sky. Perhaps more accurately, it is an object's being a certain way (e.g., the e-mail message's containing certain words in a certain order) and a property's being instantiated (e.g., the greyness of the sky) that really carries the information.

While recognizing these complications, we need some way of talking about the general case. Following Dretske, we choose to focus on claims of the form "a's being F carries the information that b is G." The main advantage of this form is that one can go on to ask what information is carried by b's being G, without committing too many grammatical offenses. A bonus is that the information carried by the Fness of a is likely to be the same as that carried by a's being F.

At first sight, Dretske's way of talking about information conveyance seems ill-suited to describing the information carried by events. There is no easy way of describing Mary's kissing John using an expression of the form "a's being F." We solve this problem by recognizing the distinction between a particular event (an event *token*) and a type of event. The occasion e of Mary kissing John on a particular day in a particular place and in a particular way is a token of the type E of kissings in which Mary is the kisser and John is the person she kissed. Thus we can talk of the information carried by e's being of type E and remain within Dretske's scheme. As will become apparent, the making of type-token distinctions is very important to our project, and the reasons for making the grammatical distinction here are not as superficial as they might at first appear.

1.5 Approaches to Information

In this section we survey a number of approaches to information flow, starting with Dretske's. In doing so, we have two motivations. One is to make clear some of the difficulties involved in answering our basic question, especially those having to do with exceptions and errors. The other is because our own proposal draws on aspects of each of these approaches. Our purpose here, then, is not to knock down straw men, but rather to pay our dues to important ideas that, although inadequate on their own, are in fact closely related to our own proposal.

Information, Probability, and Causation

The meteorologist who tells us that it will be warm and dry today in the Sahara desert is not being very informative. If she tells us that there will be three inches of rain, and she is right, then we will be properly awed.

The example suggests an inverse relationship between information and probability, namely, the less likely the message, the more informative it is. The connection is thoroughly exploited in communication theory, the quantitative theory of information used for the analysis of the efficiency of channels in communication networks.[6] Communication theorists consider a network to be composed of distinct, nondeterministic, interdependent processes, the behavior of which conforms to a probability distribution. A *channel* is a part of the network responsible for the probabilistic dependence between two of the component processes, called the *source* and the *receiver*. Quantities measuring the flow of information from source to receiver, the noise in the channel, channel capacity, and so on, can be computed from the probability distribution. The basic idea is that the amount of information associated with an event is determined by how unlikely it is to have occurred and so by the reciprocal of the probability of it occurring. Logarithms (to base 2) are taken in order to make the measure additive.

Communication theory is an established branch of engineering that has proved very useful to designers of communication devices. But its theoretical impact is far wider. Any physical system whose behavior is modeled in probabilistic terms can be regarded as an information system in which information flows according to the equations of the theory.

Impressed by the universality of the approach, a number of philosophers have tried to use the theory to elucidate our ordinary concept of information. Attractive as this appears, there are a number of obstacles. Firstly, communication

[6] The classical theory is presented in Shannon (1948). Dretske gives a summary of the relevant parts of the theory in Dretske (1981).

theory is concerned with "amounts of information ... not with the information that comes in those amounts" (Dretske, 1981, p. 3). The engineer is only interested in how much information is transmitted in a network or lost to noise, not in the details of the communication itself, which are consequently omitted from the mathematical model. Nonetheless, it may be hoped that the quantitative theory places constraints on solutions to the more philosophically interesting problem of determining what information is conveyed. Dretske adopts this view and proposes a number of such constraints.

The second obstacle is that communication theorists are interested only in averages. It is the average amount of noise, equivocation, and information transmitted that matters to the engineer, not the amount of information transmitted on a particular occasion. This is not such a serious problem, and Dretske shows that the mathematics of communication theory can be adapted to give a measure of the amounts of information involved in a particular communication between a source and a receiver. He defines the information $I(E)$ generated by an event E at the source and the amount of information $I_s(r)$ about the source process carried by the receiver process as a function of the state r of the receiver and the state s of the source.[7]

These quantities do not determine what information is carried by the signal, but they do constrain the search. If the receiver carries the information that the event E occurred at the source, then the amount of information about the source carried by the receiver must be *at least as much as* the amount of information generated by the occurrence of E at the source. Furthermore, E must have occurred, otherwise the information carried by the receiver is not information but *mis*information. Finally, Dretske claims that the information measured by $I_s(r)$ must "contain" the information measured by $I(E)$. Although lacking a precise sense for "containment" between quantities of information, he maintains that these conditions are sufficient to establish the following definition as the only viable contender:

Dretske's Information Content: To a person with prior knowledge k, r being F carries the information that s is G if and only if the conditional probability of s being G given that r is F is 1 (and less than 1 given k alone).

The role of a person's prior knowledge in determining whether information is carried is important for Dretske's epistemology. It allows him to distinguish

[7] Suppose that the source is in one of the mutually exclusive states $s_1, \ldots, s_n$ with probability $p(s_i)$. The probability $p(E)$ of E occurring at the source is given by the sum of the $p(s_i)$ for those s_i compatible with E's occurrence. Then $I(E) = \log(1/p(E)) = -\log p(E)$. Moreover, if $p(s_i \mid r)$ is the conditional probability of the source being in state s_i given that the receiver is in state r, then $I_s(r) = \sum_i p(s_i \mid r)(\log p(s_i \mid r) - \log p(s \mid r))$.

between an "internal" and an "external" contribution to assessing if something is known. The relativity of information to a person's prior knowledge is set against a background of probabilities, fixed by more or less objective features of the world.

Dretske's requirement that the conditional probability must be 1, not just very close to 1, in order for information to flow is supported by the following argument. Consider any repeatable method of sending messages. If the conditional probability that s is G given the receipt of a message is less than one, by however small a margin one cares to consider, then in principle one could resend the message using the same process so many times that the conditional probability that s is G given the final receipt of the message is close to zero. Dretske claims that if the process is good enough to transmit the information on one occasion, then no amount of duplication can prevent the information from flowing. This is captured by the following principle:

Dretske's Xerox Principle: If r being F carries the information that s is G, and s being G carries the information that t is H, then r being F carries the information that t is H.

The main objection to maximal conditional probability is that it sets too high a standard. In perception, for example, the possibility of perceptual mistakes suggests that the conditional probability of the scene being as one perceives it to be is seldom if ever as high as 1. Widespread skepticism would seem to be inevitable.[8]

Dretske has an interesting, and for our theory important, response. He points out that the relation between the probability of an event and its occurrence presupposes that certain conditions are met. The probability of a tossed coin landing heads may be 1/2, but it will definitely not land heads if it is snatched from the air in midtoss. In assessing the probability to be 1/2 we assume that such breaches of fair play do not occur – but that is not to say that they are impossible. Applying this strategy to perception, we can say that the assessment of probability for everyday perception presupposes that "normal" conditions obtain. When the stakes are higher, as they are in a murder trial for example, we may change the assessment of probability by considering a range of abnormal circumstances that may affect the veridicality of a witness's perception.

The strategy is a subtle one. When determining whether a signal carries some information, it is important for Dretske that there is a distinction between the "internal" contribution of a person's prior knowledge and the "external"

[8] On some versions of probability theory, events with zero or infinitesimally small probability may be regarded as possible. Koons (1996) has an ingenious proposal along these lines.

contribution of objective probabilities. But now it turns out that these probabilities, although objective, are relative, not absolute. In effect, Dretske admits a further parameter into the account: whether information flows depends not only on prior knowledge but also on the probability measure used. Different standards of relevant possibility and precision give different probability measures and so give different conclusions about information flow and knowledge. This relativism is downplayed in Dretske's book, but it is an unavoidable consequence of his treatment of skepticism.

Another objection, acknowledged by Dretske, is that his account makes no room for *a priori* knowledge. Consider, for example, Euclid's theorem that there are infinitely many prime numbers. If it makes any sense to talk of the probability of mathematical statements, then the probability of Euclid's theorem must be 1. Given any prior knowledge k, the conditional probability of Euclid's theorem given k is also 1 and so no signal can carry it as information. This is a serious defect that threatens any probabilistic theory of information flow.[9]

The use of probability to give an account of information is closely related to the similar use of probability to account for causation. A proposed solution to one problem often suggests a candidate for the other. Nonetheless, one must be careful to distinguish the two. Causal relations often underpin informational relations, but they are not the same. One important difference is that the direction of information flow is not necessarily aligned with the direction of causation. Present events can carry information about conditions in the remote past and about their future consequences, but there are strong arguments against the possibility of a present event causing a past event.

Moreover, a causal connection between two events, in whatever direction, is neither necessary nor sufficient for information flow. Suppose that Judith's flashlight may be switched on by either of two switches: a slider on the side of the case and a push button at the end. On a given occasion, Judith flashes the light with the push button. The occurrence of the flash was caused by the pushing of the button, but it does not carry the information that the button was pushed because the same effect could have been achieved using the slider. What about the other direction? Does the pushing of the button carry the information that the flashing occurs? Perhaps it does, in this case, but not in general. If there is a loose connection, for example, or the battery is running low, then the pushing of the button may happen to cause the light to flash without carrying the information that the light is flashing.

[9] It does no good to loosen the definition to allow unit conditional probability given the signal to be sufficient for information flow, irrespective of the conditional probability without the signal, for then *any* signal would be deemed to carry the information that there are infinitely many primes.

The above discussion shows that a causal connection, in either direction, is not sufficient for information to flow. That it is also not necessary can be seen by considering cases in which two causally unrelated events have a common cause. Suppose, for example, that two divers set their stopwatches for a thirty-minute dive. One watch going to zero carries the information that the other watch does the same, but neither event causes the other.

Information and the Elimination of Possibilities

A familiar idea in the philosophy of language is that the semantic value of a statement is given by the set of "possible worlds" in which it is true. This can be used to give an account of information content:

Possible-Worlds Information Content: To a person with prior knowledge k, r being F carries the information that s is G if in all the possible worlds compatible with k and in which r is F, s is G (and there is at least one possible world compatible with k in which s is not G).

Applying this account to epistemology, we see that it gives us a counterfactual account: whether one knows that p depends not only on the facts of this world but also on what goes on in other possible worlds.

As it stands, the account is open to the same skeptical charge leveled against Dretske in the previous section. The skeptic proposes that there is some possible world in which our perceptions are radically mistaken. The mere possibility of one such world is enough to ensure, on this definition of possible-worlds information content, that none of our perceptual beliefs carry the information they should.

Philosophers have addressed this problem by using a more sophisticated account of the semantics of counterfactual conditionals (Lewis, 1973; Stalnaker, 1984; Nozick, 1981; Dretske, 1970, 1971). The basic idea is to restrict the range of quantification from all possible worlds to all "near" or "normal" worlds, where what counts as "near" or "normal" may depend on the conditional being evaluated. Taking a cue from Dretske, we may interpret this move in terms of standards of evaluation. The statement that information flows presupposes that these standards are met. In Dretske's probabilistic account, the standards of evaluation are modeled by a probability measure. It is only in the context of an evaluation of probability that information is said to flow or not, and assessment of probability always presupposes that certain conditions are held fixed. In the theories of conditionals of Stalnaker and Lewis, we can think of what counts as a "near" possible world as modeling these standards.

Information and State Spaces

Within applied mathematics and experimental science, similar normality assumptions are made. The evaluation of some experimental data as confirming or conflicting with theory presupposes that "experimental conditions" were met. It is instructive, therefore, to examine the use of mathematical models in science, looking for the role played by this presupposition.

The most common kind of model used for studying regularities in physical systems is a "state space." A state space consists of a set Ω with some sort of mathematical structure defined on it, together with an assignment of elements of Ω to the system at different times in its evolution. The elements of Ω are called "states" because the characteristic feature of the model is that the system is deemed to be in only one state at a time.

Newtonian physics, for example, studies systems consisting of a finite number of particles in space. The complete state of the system is assumed to be determined by information about position (relative to some coordinate scheme) and velocity of each particle in the system. Because both position and velocity are determined by three magnitudes, the state of a system of n bodies can be modeled by a vector of $6n$ real numbers; that is, we can take Ω to be the vector space $\mathbb{R}^{6n}$, where $\mathbb{R}$ is the field of real numbers. The basic assumption of classical mechanics is that everything about such a system is a function of this state. The object of the enterprise is to figure out ways of writing down theories, usually in the form of some sort of equation that specifies the relationships between states of the system at different times.

State spaces are also used in computer science where computing machines are assumed to have a finite number of possible states, and that computation proceeds by means of transitions from state to state. These states are not assumed to have any *internal* structure; instead they are related by the possible transitions between them. Thus, in both examples, there is some additional mathematical structure on the set of states: a vector space in one case and a transition function in the other.

The term state *space* encourages one to regard such models using a spatial metaphor. The states of the space are thought of as locations in a landscape whose geography is molded by physical laws (as expressed by the equations). The evolution of the system is interpreted as motion in this abstract landscape, usually according to some principle of following the path of least resistance.

The construction of state spaces often proceeds by isolating certain measurable attributes, the so-called "observables" of the system, and a range of values for each observable. For example, in an experiment to investigate the behavior of gases, a scientist might decide that the pressure, volume, and temperature are the relevant observables, and that in his laboratory, these observables will take values within the ranges 1–20 bar, 0.001–2 liters, and 0–100 degrees Celsius, respectively. From these observables one can specify an initial set of states, namely, one for each assignment of values to the three attributes. The gas in the laboratory is in exactly one of these states at a time. Which state it is in can be determined by measurement.

The set of possible states is pared down by further investigation. Boyle's law tells us that only those states in which the product of the pressure and the volume is some specific multiple of the temperature are possible. This gives us a way of understanding information flow. For example, knowing that the gas is contained in a 1-liter bottle, the behavior of the thermometer carries information not just about the temperature but also about the pressure of the gas. We can define a notion of information content in a way that is parallel to the possible-worlds definition.

State-space Information Content: To a person with prior knowledge k, r being F carries the information that s is G if in every state compatible with k in which r is F, s is G (and there is at least one state compatible with k in which s is not G).

Despite formal similarities, there are several important differences from the possible-worlds version. Firstly, the definition clearly presupposes a notion of possible state, and therefore a notion of state space, according to our earlier analysis of the concept. Typically, the notion of state will be applicable to only a small part of the world for a limited duration of time. The gas experiment, for example, occurs inside a laboratory and only lasts for a few hours, at most. This is in contrast to the notion of a possible world, which is typically taken to be all-encompassing.[10]

Secondly, in the case of scientific experiments, certain "experimental conditions" must be maintained throughout the limited domain of application. For state-space models constructed by selecting observables and ranges of values, the experimental conditions are often taken to be that (i) the values of the selected observables should not exceed those in the specified ranges and (ii) all other observables (the parameters) should remain fixed throughout. The satisfaction of these conditions is a presupposition of the laws governing the system.

If during the gas experiment, the bottle containing the gas developed a slight fracture, allowing the gas to escape slowly into the surrounding room, then

[10] This is not true of Stalnaker's (1984) work. Indeed, one way of interpreting talk of possible worlds, close to Stalnaker's interpretation, is to identify possible worlds with possible states of this world.

the regularities expressed by Boyle's law would cease to be observed, and the behavior of the thermometer would no longer carry information about the pressure of the gas.

A failure of this kind is not considered a counterexample to Boyle's law; it is merely a case in which the law's presuppositions were not satisfied. For this reason, the possibility of a leak does not count against the claim that information flows when there is no leak.

We can now return to the question of how to make our definition of information content respect varying standards of evaluation. The basic idea is to apply something similar to the meeting of "experimental conditions" whenever the notion of a state space is used. For example, in describing a coin toss as having the outcomes heads and tails, we presuppose that the coin will be tossed in a "normal" manner, with no interference from participants and certainly no coin snatching in midtoss.

The example also illustrates the relationship between Dretske's definition information content and the state-space version. The rational assignment of probabilities to events presupposes a probability measure on a set of possible outcomes. If we take the set of outcomes to be our set of possible states, then we will have a state-space system within which the state-space definition of information content will be equivalent to Dretske's.

The state-space conditions of a state space constructed in this way are just those presupposed by assigning probabilities. As such, they reflect our standards of evaluation that may depend on many factors, but once fixed the resulting system of regularities determines an objective conception of information flow.

In practice, of course, it is next to impossible to pin down these conditions precisely. Often we can tell if a condition is violated; lacking any evidence that they are violated we will have reasonable confidence that they are obeyed. But the epistemologically important point is that the definition of information content (and hence of knowledge) only depends on the conditions being met, not on our knowing that they are met.[11]

In summary, recall that the First Principle of Information Flow is that information flow results from a distributed system of regularities. The question of whether information flows or not therefore presupposes a system of regularities against which it can be evaluated. In this section we have considered a model of systems of regularities using the notion of a state space. A definition of information content follows. It is formally similar to the possible-worlds definition

[11] The skeptic's hypothetical scenario shows that we *could* fail to detect that the conditions are not met, but this does not affect the issue of whether information actually flows, assuming that the conditions are in fact met.

but differs both in having a clearly limited domain of application and in clearly presupposing that certain conditions are met within that domain. It is similar to Dretske's definition but differs in not requiring a probability measure on sets of states.

Inference and Information

Inference has something crucial to do with information. On the one hand, inference is often characterized as the extraction of implicit information from explicit information. This seems reasonable, because it is clearly possible to obtain information by deduction from premises. In carrying out these inferences in real life, we take as premises some background theory of how the world works. On the other hand, getting information typically requires inference. Judith, for example, had to infer something about the mountain from her map, and Miranda had to infer something about Judith from the distress signal.

Some philosophers would go farther and maintain that the distinction between knowledge and true belief can be analyzed in terms of inferential relations alone. This "internalism" differs from the "externalist" position adopted here, according to which the world must participate in some way. On an internalist account, explanations of the sort used in discussing Judith's rescue would not be required to explain the fact that the pilot knew where to look. Still, they would be required to explain why it was that Judith was there to be found.

The close relation between inference and information suggests a different, more internalist, take on information:

Inferential Information Content: To a person with prior knowledge k, r being F carries the information that s is G if the person could legitimately infer that s is G from r being F together with k (but could not from k alone).

This proposal is refreshingly different from the earlier ones and is promising in a number of regards. First, it relativizes information to a person's ability to infer, that is, to some kind of information-processing abilities, a feature notably lacking in the other approaches.

Consider Judith and Miranda. If Judith had no knowledge of maps, she would not have been able to infer anything about where she was from the map, so the map would have carried no information to her. Similarly, if Miranda had known nothing about Morse code the outcome of the story would most likely have been quite different. Under these conditions she would not have taken the light flashes to be a signal of distress. There would have been no information flow, seemingly because there would have been no inference.

Secondly, the background theory k of the agent comes much more actively into this account. It is not just there as a parameter for weeding out possibilities; it becomes a first-class participant in the inference process. It makes Judith's, Miranda's, and the pilot's everyday knowledge of the world play key roles in the story, which it seems it should.

Finally, by relativizing information flow to human inference, this definition makes room for different standards in what sorts of inferences the person is able and willing to make. This seems like a promising line on the notorious "monotonicity problem."

In classical logic, the inference from "α entails γ" to "α and β entails γ" is valid and fundamental. This form of inference is sometimes called "Monotonicity" or "Weakening" and is symbolized as follows:

$$\frac{\alpha \vdash \gamma}{\alpha, \beta \vdash \gamma}.$$

(In this way of formulating logic, initiated by Gentzen, multiple items on the left of $\vdash$ are treated conjunctively, multiple items on the right are treated disjunctively.) After all, if α and β both hold, then α holds, so we have γ. But consider the following:

The switch being on entails that the bulb is lit.

This regularity is a commonplace bit of information about Judith's flashlight. On the earlier accounts, it would be seen as capturing some kind of regularity of the system. On the present account, it might be seen as an inference permitted by Judith's background theory k of the world.

However, this conditional has exceptions. The battery might be dead, for example. From the logical point of view, this looks like a problem because weakening seems to allow us to conclude the following from the above regularity:

The switch being on and the battery being dead entails that the bulb is lit.

The inference is clearly unwarranted and unwelcome.

The difficulty seems closely related to the problem of error in information flow. In the first three approaches, error is seen as involved with changing the probability space, changing what counts as a normal or near possible world, or changing the state space. The present proposal, it seems, might simply take the ability to recognize such shifts as a property of legitimate human inference.

Difficulties start to appear, however, when we probe more deeply into the role "legitimate" plays in this account. It is clearly crucial, because without it the person could infer anything from anything and it would count as information for that person. But if Miranda had inferred from the light flashes that they were caused by giant fireflies on the mountain, she would have been misinformed by the flashes, not informed. Similarly, had the helicopter pilot inferred from Miranda's message that she was deluded, he would have been wrong and so misinformed. Such considerations show that the inferences allowed by the definition must, at the very least, be sound inferences.

In a way, this is promising, because it suggests that this proposal has to be related to a more semantic account and so might be seen as a strengthening of one or more of the earlier proposals. But the question is, why not allow any sound inference? In particular, why not allow the sound rule of weakening?

Taking our cue from the earlier responses to problems involving exceptions, it seems that one might try to pin the problem on the relationship of the person's theory k of the world and the background conditions and standards of the person. In our above example, it seems that the premise that the battery is dead violates the background theory k used for the first inference and so causes one to change k. The account of information flow presented here relates inference and background theory with the kinds of background conditions built into the more semantic approaches.[12]

1.6 Summary

In describing the role of information in the modern world, an old word has been appropriated and its meaning gradually transformed. Yet the new uses of "information" only serve to highlight its function as a linking term in everyday explanations. Whenever a complete understanding of some series of events is unavailable or unnecessary for providing an explanation, we can make do with information talk. Whether or not these partial explanations are ultimately reducible to a single physical science, there is sufficient reason to investigate the conception of information flow on which they rest.

Information is closely tied to knowledge. Following Dretske, we think that epistemology should be based on a theory of information. The epistemic properties of an agent's mental processes should be analyzed in terms of their informational relationships to each other, to the agent's actions, and to the agent's environment.

But what is information and how does it flow? We stated the first principle of information flow: information flow results from regularities of distributed systems. The systems in which information flows are distributed because they are made up of parts related by regularities. The existence of such a system

[12] Of the theories available, the Lewis and Stalnaker theories of nearby possible worlds seem most like this sort of theory.

is presupposed by talk of the flow of information, and if we are to understand informational explanation in any more than a metaphorical way, we must find out which system is presupposed.

The central question, in a nutshell, is this: *How is it that information about some components of a system carries information about other components of the system?* We have looked for an answer in various places: in Dretske's probabilistic analysis, in the idea that information eliminates possible worlds, in the idea that information tracks possible movement in a state space, and in the connection between information and inference relative to a theory. In each case we offered a preliminary definition of information flow and a discussion of the presuppositions of the definition, especially concerning the possibility of error.

The existence of such different approaches to the study of information flow might make one wonder whether there is any unity to the subject. However, the perceived relationships between these accounts suggest they might all be seen as part of a general theory. In this book, we present what we hope is such a theory.

Lecture 2

Information Channels: An Overview

To understand the account presented here, it is useful to distinguish two questions about information flow in a given system. What information flows through the system? Why does it flow? This book characterizes the first question in terms of a "local logic" and answers the second with the related notion of an "information channel." Within the resulting framework one can understand the basic structure of information flow. The local logic of a system is a model of the regularities that support information flow within the system, as well as the exceptions to these regularities. The information channel is a model of the system of connections within the system that underwrite this information flow.

The model of information flow developed here draws on ideas from the approaches to information discussed in Lecture 1 and, in the end, can be seen as a theory that unifies these various apparently competing theories. The model also draws on ideas from classical logic and from recent work in computer science. The present lecture gives an informal overview of this framework.

2.1 Classifications and Infomorphisms

Fundamental to the notions of information channel and local logic are the notions of "classification" and "infomorphisms." These terms may be unfamiliar, but the notions have been around in the literature for a long time.

Paying Attention to Particulars

We begin by introducing one of the distinctive features of the present approach, namely its "two-tier" nature, paying attention to both types and particulars.

Suppose we are giving a state-space analysis of a system consisting of two dice that are being tossed. There are thirty-six possible states of this system,

corresponding to fact that each die can come up with one of the numbers $1, \ldots, 6$ on top. Thus the set Ω of possible states of the system is taken to be the set $\{\langle n, m\rangle \mid 1 \leq n, m \leq 6\}$. Suppose Judith is interested in the situation where the system happens to come up with a total of seven. Thus Judith is interested in the outcome being in a state in the set

$$\alpha = \{\langle 1, 6\rangle, \langle 2, 5\rangle, \langle 3, 4\rangle, \langle 4, 3\rangle, \langle 5, 2\rangle, \langle 6, 1\rangle\}$$

In probability theory, this set α is said to be the "event" of getting a seven.

This talk of events can be a bit misleading since α is not an event at all, but rather a type of event, that is, a way of classifying any particular roll of the pair of dice, or indeed, of any roll of any pair of dice that comes out with a total of seven. Whereas there are only thirty-six states, and so 2^{36} events, there are potentially an infinite number of particulars. Intuitively, the rolls of the dice are token events, α is (or models) a type of event.

Probability theory, and applied mathematics in general, works at the level of types. One considers a system that has various instances s_t at time t. Each instance s_t is assumed to be in a unique state in Ω, written state$(s_t) \in \Omega$. Sets α of states are called events. Intuitively, if state$(s_t) \in \alpha$, then s_t is of type α. Once this framework of states and events is set up, the instances themselves are usually ignored.

For a theory of information, however, these particulars, or instances, cannot be ignored. It is particulars, things in the world, that carry information; the information they carry is in the form of types. It was a particular lighting event that carried the information about Judith. That it was an event of type S.O.S. carried the information that Judith was in distress. It was a particular map that carried information (and misinformation) about the mountain. The map's being of a certain type carried information that the mountain was of some other type. We codify this observation in our second principle.

Second Principle of Information Flow: Information flow crucially involves both types and their particulars.

Notice that Dretske's account of information flow does not really respect this principle. On his analysis, information flow is characterized entirely in terms of conditional probabilities of "events." But, as we have seen, the events of probability theory are really types of events, not particular events. In spite of appearances to the contrary, there are no particulars in Dretske's analysis.

In this book we use the term "token" for the instances or particulars that carry information. For readers coming out of philosophy, this terminology may bring with it baggage from the philosophy of language that we hasten to avoid.

articular, any suggestion that tokens must have something like "syntax" is unwelcome. By a token, we mean only something that is classified; by we mean only something that is used to classify.

Classifications

der to relate the diverse approaches to information surveyed in Lecture 1, one classifies tokens by sets of states (i.e., "events") in one approach, entences in others, and by a host of other devices in still others, the theory ented here uses a very general framework, the notion of a classification. otion is a very simple one, and one that has been used in various contexts to this book.[1]

efinition. A *classification* $A = \langle A, \Sigma_A, \models_A \rangle$ consists of a set A of objects to assified, called *tokens* of A, a set Σ_A of objects used to classify the tokens, d the *types* of A, and a binary relation $\models_A$ between A and Σ_A that tells one h tokens are classified as being of which types.

classification is depicted by means of a diagram as follows.

$$\begin{array}{c} \Sigma_A \\ \Big| \models_A \\ A \end{array}$$

ample 2.1. We might classify flashlight bulbs at times, say b_t, by means of pes LIT, UNLIT, and LIVE. For example, $b_t \models$ LIT if the bulb instance b_t that is, if b is lit at time t.

ample 2.2. A familiar example of a classification is that of a first-order uage. Here the tokens are particular mathematical structures M, and the are sentences α of the language, and $M \models \alpha$ if and only if α is true in M.

ample 2.3. Another example would be that of classifying rolls of a pair of Suppose we use the set $\Omega = \{\langle n, m\rangle \mid 1 \le n, m \le 6\}$ as our state space

[1] Birkhoff dubbed such structures "polarities" in Birkhoff (1940). They are used in Hartonis Dunn (1993). The literature on formal concept analysis calls them "contexts"; see Hardegree) and, for a discussion and historical references, Chapter 11 of Davey and Priestley). More recently they have been called "Chu spaces" in theoretical computer science. arge literature has grown up around this topic, especially in the work of Vaughan Pratt his colleagues. A complete list of these papers can be found on the World Wide Web at boole.stanford.edu/chuguide.html.

Ω for analyzing these tosses. Let us suppose that the dice are chosen from a set D of six-sided dice. These dice may be tossed repeatedly. We might model particular tosses by triples like $x = \langle d_1, d_2, t\rangle$, where $d_1, d_2 \in D$ are distinct and t is the time of a particular toss. These form the tokens of a classification. Each toss x has, we assume, some state $state(x) \in \{\langle n, m\rangle \mid 1 \le n, m \le 6\}$. For example, the state of $\langle d_1, d_2, t\rangle$ is $\langle 1, 5\rangle$ if and only if d_1 lands at t with the numeral 1 showing and d_2 lands at t with the numeral 5 showing. The types of our classification are, then, the events over Ω, that is, subsets $\alpha \subseteq \Omega$, and $x \models \alpha$ if and only if the state of x is in α.

As these examples illustrate, we sometimes take the tokens of a classification to be structureless mathematical objects, while at other times we give them structure so as to relate them to other tokens of other classifications more effectively.

If a particular classification models some device (or class of devices), say Judith's whole flashlight or the World Wide Web, then the types of the classification represent all the properties of the device of relevance to our model. If the tokens of the classifications represent all the possible instances of the device deemed relevant to the problem at hand, then the classification gives rise to a "theory" of the device, namely, those relationships between types that hold for all the tokens. We want to define this theory.

In logic it is usual to take a theory to be a set of sentences, together with some kind of notion of entailment, $\Gamma \vdash \alpha$, between theories and sentences. Because we are working in a more general setting, we want to allow theories that have as premises not just sets of sentences but quite generally sets of types. Moreover, following Gentzen, we treat entailment not just as a relation between sets of sentences and single sentences, but we allow sets on both sides. In doing so, however, things work much more elegantly if one treats the set Γ on the left side of $\Gamma \vdash \Delta$ conjunctively and that on the right disjunctively. By a *sequent* we just mean a pair $\langle \Gamma, \Delta\rangle$ of sets of types. The reader not familiar with logic may want to restrict attention to the case where there is a singleton set on the right-hand side, for now.

Definition. Let A be a classification and let $\langle \Gamma, \Delta\rangle$ be a sequent of A. A token a of A *satisfies* $\langle \Gamma, \Delta\rangle$ provided that if a is of type α for every $\alpha \in \Gamma$ then a is of type α for some $\alpha \in \Delta$. We say that Γ *entails* Δ in A, written $\Gamma \vdash_A \Delta$, if every token a of A satisfies $\langle \Gamma, \Delta\rangle$. If $\Gamma \vdash_A \Delta$ then the pair $\langle \Gamma, \Delta\rangle$ is called a *constraint* supported by the classification A.

The set of all constraints supported by A is called the complete theory of A and is denoted by Th(**A**). The complete theory of A represents all the regularities supported by the system being modeled by A.

Example 2.4. In Example 2.1 we get constraints like LIT ⊢ LIVE, since every lit bulb is live, and LIT, UNLIT ⊢. (Strictly speaking, we should be writing these as {LIT} ⊢ {LIVE} and {LIT, UNLIT} ⊢, but we leave out the set brackets because no confusion is likely.) The latter constraint says that no bulb is both lit and unlit, because it says that any bulb of both types on the left is of at least one type on the right, but there are no types on the right. Similarly, we get ⊢ LIT, UNLIT because every bulb is either lit or unlit.

In Example 2.2, we have $\Gamma \vdash \Delta$ if and only if every structure M that is a model of every sentence in Γ is a model of some sentence in Δ. This is just the classical notion of logical entailment from first-order logic.

In Example 2.3, suppose that each state $\langle n, m\rangle$ in every event in Γ is in at least one event in Δ. Then for any toss x, if it satisfies every type in Γ, it must also satisfy some type in Δ, and so $\Gamma \vdash \Delta$. Whether or not the converse holds will depend on whether the set of tokens of the classification contains, for each state $\langle n, m\rangle$, a toss with $\langle n, m\rangle$ as its state. If so, then the converse holds. But if not, then intuitively the theory of our classifications will capture some accidental generalizations. This is a theme we will be coming back to.

Here are five special kinds of constraint to keep in mind.

Entailment: A constraint of the form $\alpha \vdash \beta$ (the left- and right-hand sides are both singletons) represents the claim that α entails β.

Necessity: A constraint of the form $\vdash \alpha$ (the left-hand side is empty, the right is a singleton) represents the claim that the type α is necessarily the case, without any preconditions.

Exhaustive cases: A constraint of the form $\vdash \alpha, \beta$ (the left-hand side is empty, the right is a doubleton) represents the claim that every token is of one of the two types α and β, again without any preconditions.

Incompatible types: A constraint of the form $\alpha, \beta \vdash$ (the right-hand side is empty, the left is a doubleton) represents the claim that no token is of both types α and β. (This is because no token could satisfy any type on the right, because there are none, and hence could not satisfy both types on the left.)

Incoherent types: A constraint of the form $\alpha \vdash$ (the right-hand side is empty, the left is a singleton) represents the claim that no token is of types α.

Infomorphisms Between Classifications

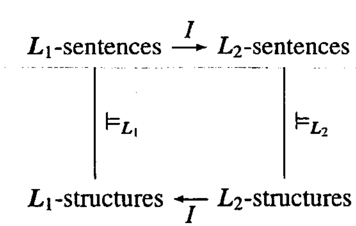
In modeling a distributed system, one uses a classification for each of the components and another for the system as a whole. Consequently, we need a way to model the relationship between the whole and its parts.

As an analogy to guide us, let us think about the example of number theory considered as a part of set theory. Applying Example 2.2, suppose that L_1 is the language of arithmetic, with numerals 0 and 1 and additional nonlogical symbols like $<, +, \times, =$, and so on. By the tokens of L_1 we mean any structure that satisfies the basic axioms PA of Peano arithmetic; the types are sentences formulated using the above symbols plus standard symbols from logic. Let L_2 be the language of set theory, with only $\in$ and $=$ as nonlogical symbols. By the tokens of L_2 we mean any structure that satisfies the usual axioms ZFC of Zermelo-Fraenkel set theory; again types are sentences formulated in terms of $\in$, $=$, and the basic symbols of logic.

One of the standard themes in any course on set theory is to show how to translate number theory into set theory using the finite von Neumann ordinals. Formally, what is going on is the development of an "interpretation." One shows how to translate any sentence α of number theory into a sentence α^I of set theory.

At the level of structures, though, things go the other way. A model of number theory does not determine a unique model of set theory. Indeed, some models of number theory are not parts of any model of set theory at all, because set theory is much stronger than number theory.[2] By contrast, any model V of set theory does determine a unique model $N = V_I$ of number theory. The reversal of directions is quite important.

This example is a special case of the notion of an interpretation (sometimes called a translation) of one language into another. There are two aspects to an interpretation, one having to do with tokens (structures), the other with types (sentences). An interpretation $I : L_1 \rightleftarrows L_2$ of language L_1 into language L_2 does two things. At the level of types, it associates with every sentence α of L_1 a sentence α^I of L_2, its "translation." At the level of tokens, it associates with every structure M for the logic L_2 a structure M_I for the logic L_1. This might be diagramed as follows:



[2] If the model of number theory is one where some theorem of set theory about numbers, like Gödel's formal encoding of the consistency of PA, is false, then it cannot have an associated model of set theory.

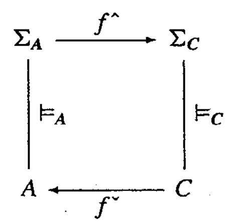
One needs to make sure that the translation α^I of α means the right thing, that is, that the translation α^I says about a structure M what α itself says about the corresponding structure M_I. Hence one requires that

$$M_I \models_{L_1} \alpha \quad \text{iff} \quad M \models_{L_2} \alpha^I$$

for all structures M for L_2 and all sentences α of L_1.

Thinking of number theory as a "part of" set theory suggests that this sort of picture might apply more generally to distributed systems. The notion of an infomorphism $f : \boldsymbol{A} \rightleftarrows \boldsymbol{C}$ gives a mathematical model of the *whole-part relationship* between instances of a whole, as modeled by a classification $\boldsymbol{C}$, and that of a part, as modeled by a classification $\boldsymbol{A}$. This is obtained by generalizing the notion of interpretation from logics to arbitrary classifications in the natural way.

Definition. If $\boldsymbol{A} = \langle A, \Sigma_A, \models_A \rangle$ and $\boldsymbol{C} = \langle C, \Sigma_C, \models_C \rangle$ are classifications then an *infomorphism* is a pair $f = \langle f^{\wedge}, f^{\vee} \rangle$ of functions



satisfying the analogous biconditional:

$$f^{\vee}(c) \models_A \alpha \quad \text{iff} \quad c \models_C f^{\wedge}(\alpha)$$

for all tokens c of $\boldsymbol{C}$ and all types α of $\boldsymbol{A}$.

Such infomorphism is viewed as a "morphism" *from* $\boldsymbol{A}$ *to* $\boldsymbol{C}$. The displayed biconditional will be used many times throughout the book. It is called the *fundamental property of infomorphisms*.

Think of the classification $\boldsymbol{C}$ on the right as a scientific classification of the tokens (instances) of the whole system, say Judith's flashlight, and the classification $\boldsymbol{A}$ on the left as a classification of tokens on one of the components of the system, say the bulb. The latter might be a scientific classification of the sort used by an engineer in designing the bulb or a more common sense classification of the sort Judith would use. By an instance of the flashlight we simply mean the flashlight at a given instant, rather than the flashlight as a type or as an object that persists through time. The infomorphism f has two parts. The lower part $f^{\vee}$ (read "f-down") assigns to each instance c of the whole flashlight the instance $a = f^{\vee}(c)$ of the bulb at the same time. The upper part $f^{\wedge}$ (read "f-up") assigns to each type α of the component classification $\boldsymbol{A}$ some "translation" $f^{\wedge}(\alpha)$ in the "language" of the classification $\boldsymbol{C}$. For example, the classification of the bulb might be very simple, with types {LIT, UNLIT, LIVE}, and the classification of the flashlight might be quite complex, in the language of electrical engineering or even basic atomic physics. The biconditional in the definition of infomorphism insures that the translation of a type α says about the whole system what α says about the component.

The use of "infomorphism" for this notion is new here, but infomorphisms are not new. In the case of logical languages, they are, as we have indicated, just the old notion of interpretation of languages. In more a general setting, these infomorphisms are known in computer science as Chu transformations. They have been studied extensively by Chu, Barr, Pratt, and others. The present authors rediscovered them in a roundabout way, trying to solve the problems of information flow discussed above. So one could look at this book as an application of Chu spaces and Chu transformations to a theory of information.[3]

A great deal is known about classifications and their infomorphisms. For our purposes, the most important fact is that they can be combined by what is known as a "(co)limit" construction.[4] A special case of this construction allows one to "add" classifications. Given two (or more) classifications $\boldsymbol{A}$ and $\boldsymbol{B}$, these classifications can be combined into a single classification $\boldsymbol{A}+\boldsymbol{B}$ with important properties. The tokens of $\boldsymbol{A}+\boldsymbol{B}$ consist of pairs $\langle a, b \rangle$ of tokens from each. The types of $\boldsymbol{A}+\boldsymbol{B}$ consist of the types of both, except that if there are any types in common, then we make distinct copies, so as not to confuse them. For example, if $\boldsymbol{A}$ and $\boldsymbol{B}$ both had contained the type LIT, then $\boldsymbol{A}+\boldsymbol{B}$ would have two types, $\text{LIT}_{\boldsymbol{A}}$ and $\text{LIT}_{\boldsymbol{B}}$. A pair $\langle a, b \rangle$ would be of type $\text{LIT}_{\boldsymbol{A}}$ in $\boldsymbol{A}+\boldsymbol{B}$ if and only if a was of type LIT in $\boldsymbol{A}$. Thus the classification $\boldsymbol{A}+\boldsymbol{B}$ gives us a way to compare types that classify tokens from both classifications.

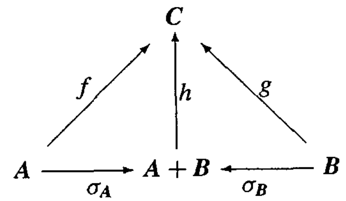
This construction works nicely with infomorphisms as well. First of all, there are natural infomorphisms $\sigma_A : \boldsymbol{A} \rightleftarrows \boldsymbol{A}+\boldsymbol{B}$ and $\sigma_B : \boldsymbol{B} \rightleftarrows \boldsymbol{A}+\boldsymbol{B}$ defined as follows:

1. $\sigma_A(\alpha) = \alpha_A$ (the $\boldsymbol{A}$-copy of α) for each $\alpha \in \text{typ}(\boldsymbol{A})$,
2. $\sigma_B(\beta) = \beta_B$ for each $\beta \in \text{typ}(\boldsymbol{B})$, and
3. for each pair $\langle a, b \rangle \in \text{tok}(\boldsymbol{A}+\boldsymbol{B})$, $\sigma_A(\langle a, b \rangle) = a$ and $\sigma_B(\langle a, b \rangle) = b$.

[3] A good deal of the mathematical content of Lectures 4–6 is already known to the people who study Chu spaces. In particular, the existence of colimits was established by Chu in his appendix to Barr (1979). We hope they will be interested in how this material fits into the general theory of information developed here.

[4] We will work through this construction in Lecture 6, where we will see that it is intimately tied up with Dretske's Xerox principle.

More importantly, given any classification C and infomorphisms $f : A \rightleftarrows C$ and $g : B \rightleftarrows C$, there is a unique infomorphism $h = f + g$ such that the following diagram commutes. (Saying that a diagram commutes means you can go either way around the triangles and get the same result.)



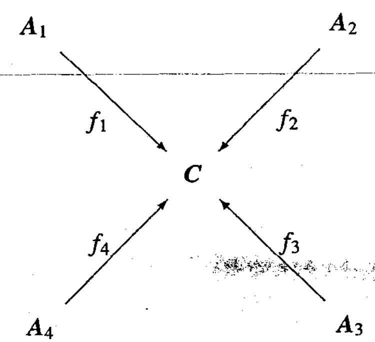
Each of the arrows represents an infomorphism and hence a pair of functions, one on types and another on tokens that goes in the opposite direction. The definition of h is obvious once you think about it. On tokens, $h(c) = \langle f(c), g(c)\rangle$. On types of the form α_A, h gives $f(\alpha)$. On types of the form α_B, use g.

2.2 Information Channels

One of the two central notions of the present approach, that of an information channel, can now be explained. Suppose there is a distributed system, modeled by means of a classification C, and several components, modeled by means of classifications A_i for i in some index set I. Because A_i is a part of C, there must be an infomorphism $f_i : A_i \rightleftarrows C$, one for each $i \in I$, reflecting the part-whole relationships between the system and its parts.

Definition. An *information channel* consists of an indexed family $\mathcal{C} = \{f_i : A_i \rightleftarrows C\}_{i \in I}$ of infomorphisms with a common codomain C, called the *core* of the channel.

An information channel for a distributed system with four components, like Judith's flashlight, can be diagrammed as follows:



Again, each of the arrows represents an infomorphism, hence a pair of functions, one on types going in and another that goes in the opposite direction, taking each token c to its ith component $a_i = f_i(c)$.

The following simple observation is at the heart of the theory developed in this book. Think of the token c of the whole system as a "connection" between its various components, the various a_i. For example, a bulb at a time is connected by a switch at a time if they are connected by being parts of, say, a single flashlight. It is this connection that allows one to carry information about the other.

Third Principle of Information Flow: It is by virtue of regularities among connections that information about some components of a distributed system carries information about other components.

The classification C and its associated Th(C) give us a way to model these regularities. Using the constraints of Th(C) and the infomorphisms, we can capture the basic principles of information flow relating the components.

It is now clear why the first two principles are so important. The first focuses attention on distributed systems, the second on their tokens, which include connections. Without the tokens, we would not be able to track which things are connected to which others and so would have no idea how to tell what is carrying information about what.

This basic point can be illustrated with a simple example. If you flip a light switch, which light comes on depends on which light the switch is connected to. Similarly, suppose you have two photos, one each of two individuals. One photo carries information about one person, the other about the other person. But what determines which person a given photo carries information about is the connection between the person and his or her own photo, a connection rooted in the process of photography.

Given the notion of an information channel, we can now propose our analysis of information flow. In this introduction, we treat the simplest nontrivial case, that of two components, a and b, of a system.

Initial Proposal. Suppose that the token a is of type α. Then a's being of type α *carries the information that b is of type β*, relative to the channel $\mathcal{C}$, if a and b are connected in $\mathcal{C}$ and if the translation α' of α entails the translation β' of β in the theory Th(C), where C is the core of $\mathcal{C}$.[5]

[5] If we compare this proposal to Dretske's probability proposal, we see one thing that Dretske has built in that we have ignored. Namely, in giving this definition one might want to exclude the case where β is a universal type, one that holds of all tokens.

Example 2.5. Here is a simple example using Judith's flashlight. Suppose $f : B \rightleftarrows F$ and $g : S \rightleftarrows F$ represent the part-whole relations between flashlights (tokens of F) classified scientifically, with bulbs (B) and switches (S) classified in commonsense ways. Let us suppose the classification F supports the constraint

$$g(\text{ON}) \vdash_F f(\text{LIT}),$$

where the antecedent ON is the type of S of switches that are on. In other words, we suppose that the flashlights that make up the tokens of F are all working normally. Then if switch s_t is connected to bulb b_t by some flashlight f_t in F, and b_t is on, this will carry the information that s_t is lit.

Let's examine a couple of the virtues of this proposal.

Veridicality

The proposal agrees with Dretske's in that it makes information veridical. That is, if a is of type α and this carries the information that b is of type β, then b is of type β. The reason is that the connection c between a and b must satisfy the constraint $\alpha' \vdash_C \beta'$ on the one hand, and it must be of type α' because a is of type α by the fundamental property of infomorphisms. Hence c must also be of type β'. But then b must be of type β, again by the fundamental property of infomorphisms.

The Xerox Principle

Dretske's Xerox principle also falls out of this proposal. If a being of type α carries the information that b is of type β, then it does so by virtue of some information channel C_1. Similarly, if b being of type β carries the information that d is of type δ, then it does so by virtue of a channel C_2. Let C be the limit of these channels. It turns out, as we will see, that this is exactly the channel we need to see that a being of type α carries the information that d is of type δ by virtue of the channel C. In other words, composing information channels amounts to taking their limit.

Shortcomings

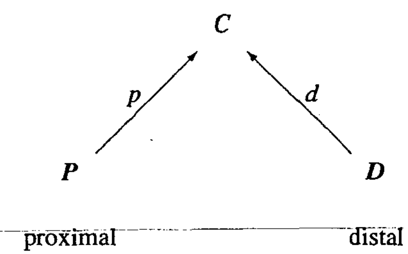
By way of introducing the notion of a local logic, we point out a couple of shortcomings with our initial proposal, one aesthetic, one substantial. The aesthetic problem is that the proposal does not directly identify the regularities on the components of the system. Rather, it characterizes them indirectly in terms of constraints on the core and translations into this core. We would like to have the regularities among the components themselves as constraints.

More seriously, our initial proposal presupposes that we have complete information about the regularities on the core of our channel, that is, the complete theory Th(C) of C. The proposal gives a God's eye analysis of information flow. But in actual fact, we seldom (if ever) have complete information about the core of a real-world channel. Usually, we have at best some kind of commonsense theory of, or some scientific model of, the core of our channel; we use it to attribute information about one component to another component. We will have a more general theory of information if we relativize our proposal in a way that permits less than perfect information about the core.

2.3 Local Logics

In trying to model commonsense theories, artificial intelligence (AI) researchers have felt the need to introduce a variety of nonclassical logics to model the way people reason about these systems. Ordinary reasoning is not logically perfect; there are logical sins of commission (unsound inferences) and omission (inferences that are sound but not drawn). Modeling this, AI has had to cope with logics that are both unsound and incomplete. These are the sorts of logics we need in order to model our less than perfect information about the core of a channel.

We can see how unsound and incomplete logics arise in reasoning about distributed systems. We give two examples, each of which can be pictured as by the following diagram:



This is a channel involving a proximal classification P, a distal classification D, and connecting classification C.

Example 2.6. For a dramatic example, consider a nuclear reactor. Such a reactor is indeed a very distributed system. An engineer operating such a system is forced to reason about what is going on in the reactor's core from information available to him at the periphery, so to speak, from various monitors, gauges, warning lights, and so forth. In this case, the proximal classification would be a classification of the control room with its monitors, gauges, and so forth. The distal classification would classify the core of the reactor. The connecting classification classifies the whole reactor. (Our terminology "core" is potentially misleading in this example, as the core of our channel is the whole reactor, not its core in the usual sense.)

Example 2.7. For a different sort of example, imagine some real-world phenomenon being studied by scientists using mathematical tools. The proximal classification is the classification of various mathematical objects. The distal classification is that of the real-world phenomenon. The connecting classification is the particular practice of mathematical modeling being employed. Thus in this example we are interested in how reasoning about the mathematical objects can give us information about the real-world phenomenon.

In both examples, we are interested in seeing what kind of a theory of the distal classification is available to someone with complete knowledge of the proximal classification. The diagram suggests breaking this problem into two parts, that of going from $\boldsymbol{P}$ to $\boldsymbol{C}$ and that of going from $\boldsymbol{C}$ to $\boldsymbol{D}$. Notice that the first step is in the direction of the infomorphism p, whereas the second step is against the direction of the infomorphism d.

We can discuss both steps at once by considering arbitrary classifications $\boldsymbol{A}$ and $\boldsymbol{B}$ and an infomorphism $f:\boldsymbol{A}\rightleftarrows\boldsymbol{B}$. Imagine someone who needs to reason about tokens on either side by using the natural theory of the other side. (In the above, f could be either $p:\boldsymbol{P}\rightleftarrows\boldsymbol{C}$ or $d:\boldsymbol{C}\rightleftarrows\boldsymbol{D}$.) Let's write Γ^f for the set of translations of types in Γ when Γ is a set of types of $\boldsymbol{A}$. If Γ is a set of types of $\boldsymbol{B}$, write Γ^{-f} for the set of sentences whose translations are in Γ.

Consider, now, the following "rules of inference": The first of these is a stylized way of saying that from the sequent $\Gamma^{-f}\vdash_{\boldsymbol{A}}\Delta^{-f}$ we can infer the sequent $\Gamma\vdash_{\boldsymbol{B}}\Delta$; the second is read similarly.

$$f\text{-}\mathbf{Intro}: \frac{\Gamma^{-f}\vdash_{\boldsymbol{A}}\Delta^{-f}}{\Gamma\vdash_{\boldsymbol{B}}\Delta}$$

$$f\text{-}\mathbf{Elim}: \frac{\Gamma^{f}\vdash_{\boldsymbol{B}}\Delta^{f}}{\Gamma\vdash_{\boldsymbol{A}}\Delta}$$

The first rule allows one to move from a sequent of $\boldsymbol{A}$ to a sequent of $\boldsymbol{B}$, whereas the second allows one to go the other way around. These inference rules have very important properties.

Let us think of them in terms of the interpretation of Peano arithmetic (PA) in set theory in Section 2.1. The first rule would allow us to take any valid sequent of PA and translate it into a sequent about models of set theory. The second rule goes the other way around, allowing us to take a sequent in the language of set theory that happens to be the translation of a sequent from number theory and allowing us to infer the latter from the former.

Consider whether these inference rules preserve validity and nonvalidity. Let us think first about whether the rules preserve validity, that is, lead from constraints of $\boldsymbol{A}$ to constraints of $\boldsymbol{B}$.

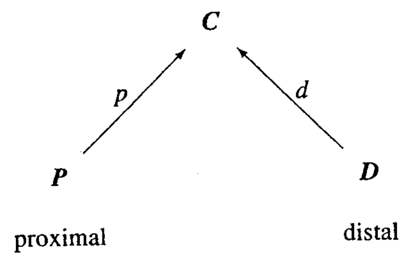
The rule f-**Intro** preserves validity. That is, if the sequent $\Gamma^{-f}\vdash_{\boldsymbol{A}}\Delta^{-f}$ is valid in $\boldsymbol{A}$, then $\Gamma\vdash_{\boldsymbol{B}}\Delta$ is valid in $\boldsymbol{B}$. This follows immediately from the fundamental property of infomorphisms. If c were a counterexample to the latter sequent, then $f(c)$ would be a counterexample to the former.

The rule f-**Elim** does not preserve validity. It is perfectly possible to have a constraint $\Gamma^f\vdash_{\boldsymbol{B}}\Delta^f$ of $\boldsymbol{B}$ such that $\Gamma\vdash_{\boldsymbol{A}}\Delta$ has a counterexample. However, no such counterexample can be of the form $f(c)$ for a token c of $\boldsymbol{B}$, again by the fundamental property of infomorphisms. In other words, the rule is sound as long as one restricts attention to tokens that come from tokens of the classification $\boldsymbol{B}$. In the case of number theory and set theory, this tells us that theorems of set theory in the language of number theory are reliable as long as we are only interested in models of number theory that are parts of models of set theory. On other models, these theorems are unreliable.

Let us turn now from preservation of validity to preservation of nonvalidity. If the premise sequent of the rule f-**Intro** is not a constraint of $\boldsymbol{A}$, can we be sure that the conclusion is not a constraint of $\boldsymbol{B}$? No, not in general. In the case of number theory and set theory, for example, there are many nontheorems about numbers that can be proven in ZFC, Gödel's consistency statement for PA being the most famous. By contrast, the rule f-**Elim** is much better in this regard. If the premise is a nonconstraint of $\boldsymbol{B}$ then the conclusion will be a nonconstraint of $\boldsymbol{A}$.

Summarizing the above observations, the rule of f-**Intro** preserves validity but not nonvalidity, whereas the rule of f-**Elim** preserves nonvalidity but not validity. Using the f-**Intro** rule, any constraint that holds for a component translates into a constraint about the system. Using the f-**Elim** rule, any constraint about the whole system gives one a constraint about the components but only guarantees that it holds on those tokens that really are the components of some token of the whole system. All bets are off on any other tokens. This is certainly a reasonable rule of inference, even if it might lead one astray were one to apply it to something that was not a component of the system, that is, even though it is not sound.

Now lets return to apply these considerations to the channel depicted earlier:



We wanted to know what happens when we use the complete theory of the proximal classification $\boldsymbol{P}$ to reason about the distal classification $\boldsymbol{D}$. We have seen that p-**Intro** preserves validities, so the theory we obtain on $\boldsymbol{C}$ is sound, but it may not be complete. That is, there may be constraints about $\boldsymbol{C}$ that we miss. On the other hand, using d-**Elim** means that we lose our guarantee that the resulting theory is either sound or complete. A sequent about distal tokens obtained from a constraint about proximal tokens in this way is guaranteed to apply to distal tokens that are connected to a proximal token in the channel, but about other distal tokens we have no guarantees.

The concept of a local logic tracks what happens when we reason at a distance in this way.

Definition. A *local logic* $\mathfrak{L} = \langle \boldsymbol{A}, \vdash_{\mathfrak{L}}, N_{\mathfrak{L}} \rangle$ consists of a classification $\boldsymbol{A}$, a set $\vdash_{\mathfrak{L}}$ of sequents (satisfying certain structural rules) involving the types of $\boldsymbol{A}$, called the *constraints* of $\mathfrak{L}$, and a subset $N_{\mathfrak{L}} \subseteq \boldsymbol{A}$, called the *normal tokens* of $\mathfrak{L}$, which satisfy all the constraints of $\vdash_{\mathfrak{L}}$. A local logic $\mathfrak{L}$ is *sound* if every token is normal; it is *complete* if every sequent that holds of all normal tokens is in the consequence relation $\vdash_{\mathfrak{L}}$.

This is the promised generalization of the notion of the complete theory of a classification. Sound and complete local logics are really nothing more or less than classifications. But infomorphisms allow us to move local logics around from one classification to another, in ways that do not preserve soundness and completeness. Given any infomorphism $f : \boldsymbol{A} \rightleftarrows \boldsymbol{B}$ and a logic $\mathfrak{L}$ on one of these classifications, we obtain a natural logic on the other. If $\mathfrak{L}$ is a logic on $\boldsymbol{A}$, then $f|\mathfrak{L}|$ is the logic on $\boldsymbol{B}$ obtained from $\mathfrak{L}$ by f-Intro. If $\mathfrak{L}$ is a logic on $\boldsymbol{B}$, then $f^{-1}|\mathfrak{L}|$ is the logic on $\boldsymbol{A}$ obtained from $\mathfrak{L}$ by f-Elim.

For any binary channel $\mathcal{C}$ as above, we define the local logic $\text{Log}_{\mathcal{C}}(\boldsymbol{D})$ on $\boldsymbol{D}$ induced by that channel as

$$\text{Log}_{\mathcal{C}}(\boldsymbol{D}) = d^{-1}[p[\text{Log}(\boldsymbol{P})]],$$

where $\text{Log}(\boldsymbol{P})$ is the sound and complete logic of the proximal classification $\boldsymbol{P}$. This logic builds in the logic implicit in the complete theory of the classification $\boldsymbol{P}$. As we have seen, $\text{Log}_{\mathcal{C}}(\boldsymbol{D})$ may be neither sound nor complete.

In Lecture 15 we will show that *every local logic on a classification $\boldsymbol{D}$ is of the form $\text{Log}_{\mathcal{C}}(\boldsymbol{D})$ for some binary channel.* Moreover, the proximal classification of $\mathcal{C}$ has a very intuitive interpretation as an idealization of $\boldsymbol{D}$. This result shows that local logics are very naturally associated with information channels.

We can now go back and generalize our initial proposal concerning information flow. That proposal presupposed knowledge of the complete theory of the core of the channel. We really want to allow the core of the channel to be thought of as the distal classification of some other channel and use the proximal channel to reason about the core and so about the components. Or, equivalently in view of the previous paragraph, we relativize our account of information flow along a channel relative to some local logic $\mathfrak{L}$ on its core.

We mentioned two problems with our initial proposal, one aesthetic, one substantial. We have dwelt at length on the substantial problem, that of complete knowledge of the theory of the core classification. We now turn to the first problem, the fact that we did not end up with constraints about the components of the system, only about their translations into the types of the core. We can solve this problem quite easily with the tools we have developed.

The existence of sums of classifications allows one to turn any information channel into one with a single infomorphism; to do so you simply take the sum of the component classifications and a corresponding sum of the component infomorphisms. If we have any channel $\mathcal{C} = \{f_i : \boldsymbol{A}_i \rightleftarrows \boldsymbol{C}\}_{i \in I}$, we can represent it by means of a single infomorphism by simply taking the sum $\boldsymbol{A} = \sum_{i \in I} \boldsymbol{A}_i$ of the $\boldsymbol{A}_i$ and the corresponding sum $f = \sum_{i \in I} f_i$ of the f_i:

$$f : \boldsymbol{A} \rightleftarrows \boldsymbol{C}.$$

Given any logic $\mathfrak{L}$ on the core, we can use the rule f-Elim to obtain a local logic $f^{-1}|\mathfrak{L}|$ on $\boldsymbol{A}$. It is this logic, with its constraints and its set of normal tokens, that captures the information flow inherent in the channel. Or, to return to the questions posed at the start of this lecture, the local logic $f^{-1}|\mathfrak{L}|$ is the "what" of information flow, the channel is the "why". The closer the logic $\mathfrak{L}$ agrees with the complete theory of $\boldsymbol{C}$, the better an account we will get of the actual information flow of the system.

Example 2.8. Let's reexamine the flashlight example in light of this discussion. Recall that $f : \boldsymbol{B} \rightleftarrows \boldsymbol{F}$ and $g : \boldsymbol{S} \rightleftarrows \boldsymbol{F}$ represent the part-whole relations between flashlights (tokens of $\boldsymbol{F}$) classified scientifically, and bulbs ($\boldsymbol{B}$) and switches (S) classified in commonsense ways. Putting these together, we obtain an infomorphism $h = f + g$ from $B + S$ to F. On flashlight tokens x, $h(x) = \langle f(x), g(x) \rangle$, that is, $h(x)$ gives us the bulb and switch connected by x. On types, h is the (disjoint) union of f and g.

Let us suppose that the classification B supports the constraint:

$$\text{LIT} \vdash_B \text{LIVE}.$$

It is easy to verify that this will also be a constraint in $B + S$. Now, whatever the classification F of flashlights is and whatever our infomorphism h does to these types, we see that

$$h(\text{LIT}) \vdash_F h(\text{LIVE})$$

must be the case. We know from the validity of the one sequent that the other must be valid.

To go the more interesting way around, suppose again that the classification F supports the constraint

$$h(\text{ON}) \vdash_F h(\text{LIT}),$$

where the antecedent ON is the type of S of switches that are on. In other words, we suppose that the flashlights that make up the tokens of F are all working normally. In this case, we obtain the sequent

$$\text{ON} \vdash_{B+S} \text{LIT}.$$

This sequent, nice as it is, is not valid! There are pairs $\langle b, s \rangle$ of switches and bulbs such that s is on but b is not lit. Not surprisingly, it only holds of those pairs that happen to be connected by a token of a single flashlight f_t of F. The pairs that are so connected are the normal tokens of our the logic obtained by h-Elim. (Notice that this is entirely parallel to the case of number-theoretic theorems of ZFC not holding of all models of PA.) This logic gives us a mathematical characterization of the information flow between bulbs and switches made possible by the channel in question.

2.4 Exceptions

We now turn to the most challenging aspect of a theory of information. As explained earlier, an important motivation for developing the details of the theory presented in this book is the severe tension between the reliability and the fallibility of constraints describing distributed systems and the consequent information flow.

Exceptions and Information Channels

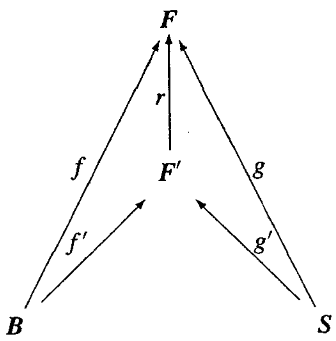
On the present account, distributed systems are modeled by means of classifications and infomorphisms. Typically, there are many ways to analyze a particular system as an information channel. Take Judith's flashlight, for example. In the first place, it is up to us as theorists to decide on the level of granularity of analysis. Do we consider just the switch and bulb, or do we want to also take into account the case, the batteries, the spring, the wires, and all the other pieces. In the second place, it is up to us to decide on the types to be used in the analysis. We might have a very simple system or a much more elaborate one. This is true of both the system as a whole and of the constituent parts. Last but far from least, there is the matter of what tokens are present in the classifications. Do we take only tokens of the particular system that happen to have occurred in the past, or do we also allow more idealized tokens to represent possibilities that may not have been realized yet? Do we admit only "normal" instances of the flashlight, where the batteries are working and there is no short circuit, or do we admit other tokens? The decisions here all go into determining what counts as a valid constraint of the classification. We summarize this as our final principle.

Fourth Principle of Information Flow: The regularities of a given distributed system are relative to its analysis in terms of information channels.

This principle includes, among other things, the key insight in Dretske's response to sckepticism, discussed in Lecture 1. Recall that that reply argued that whether or not something counted as a relevant possibility was not an absolute matter, but was dependent on some contextually based standards. In the same way, an analysis of a system as a channel is similarly dependent on contextually based perspectives and standards. Whether or not some sequent is a constraint of a given system is not an absolute matter, but depends on the channel one uses in understanding the system. Such a channel may be explicitly described by a theorist, or it may be implicit in the judgments of a user of the system. But the point is that if one changes the channel, one typically gets different constraints and so different information flow. Thus whether or not a particular token carries information about another token depends in part on the channel under consideration.

Intuitively, the more "refined" the channel used to analyze a system, the more reliable the information flow. But what sort of relation between channels is this notion of being more refined?

Example 2.9. Consider, for simplicity, two analyses of Judith's flashlight in terms of channels $\mathcal{F}$ and $\mathcal{F}'$ with core classifications F and F', respectively. If there is an infomorphism $r : F' \rightleftarrows F$ such that the following diagram commutes, then $\mathcal{F}'$ is said to be a *refinement* of $\mathcal{F}$.



Let us see why this is a reasonable definition.

Consider the rules of inference r-**Intro** and r-**Elim** for the infomorphism r, as discussed earlier. Any constraint of F' will yield a constraint about F using r-**Intro**. This means that any sequent that holds relative to $\mathcal{F}'$ will continue to hold relative to $\mathcal{F}$. However, some sequents that are constraints relative to $\mathcal{F}$ may not be constraints relative to $\mathcal{F}'$, owing to the unsoundness of r-**Elim**.

Let us assume that $\mathcal{F}'$ contains as tokens all actual instances of Judith's flashlight, including those where the batteries are dead. (It might even include wider tokens, so as to take account of whether the flashlight is in air or water, say.) Suppose, however, that $\mathcal{F}$ is just like $\mathcal{F}'$ except that it does not contain any tokens where there are dead batteries. Observe that $\mathcal{F}'$ is a refinement of $\mathcal{F}$; take r to be identity on both types of $\mathcal{F}'$ and tokens of $\mathcal{F}$.

Now suppose Judith's flashlight does indeed have a dead battery. Then b, the bulb, is connected to the switch s by a connection c' in $\mathcal{F}'$ but not relative to any connection in $\mathcal{F}$. Relative to $\mathcal{F}$ there is a constraint that the switch being on involves the bulb being lit. This is not a constraint relative to $\mathcal{F}'$, however, Judith's flashlight is an exception relative to $\mathcal{F}$ but not relative to $\mathcal{F}'$.

Now let us move back to the normal case. Suppose that switch s and bulb b are connected by being components of a normal flashlight y, one where the batteries are in working order. Does s being on carry the information that b is lit or not? Well, it depends. It does relative to the channel with core F but not with respect to the more refined channel F'.

The rule of Weakening

$$\frac{\alpha \vdash \gamma}{\alpha, \beta \vdash \gamma}$$

is perfectly valid, as long as one is working with constraints relative to a single information channel. The trouble with the inference from

The switch being on entails the bulb lighting

to

The switch being on and the battery being dead entails the bulb lighting

is that the added antecedent implicitly changes the channel relative to which the sequent is assessed; it does so by changing the standards through raising the issue of whether the battery is dead.

Another way to put it is as follows. The inference is *not* a counterexample to the valid rule of Weakening but rather an application of the invalid rule r-Elim, where $r : F' \rightleftarrows F$ is the refinement described above. The sequent

$$h(\text{ON}) \vdash_F h(\text{LIT})$$

is valid, but the sequent

$$h'(\text{ON}) \vdash_{F'} h'(\text{LIT})$$

(obtained from the former sequent by r-Elim, where $h' = f' + g'$) is not valid; there are tokens in F' that are not in F.

From the present perspective, the reluctance of people to use Weakening in such circumstances does not show that they use nonclassical logic, but rather shows that people are good at changing channels when exceptions arise, reassessing a situation in terms of a refined channel, and that this is a useful way to think about exceptions.

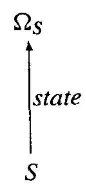
This suggests that, corresponding to the notion of refinements among channels, there ought to be a notion of refinement in local logics, and there is. More refined logics have fewer constraints but more normal tokens to which they apply.

It is too soon to tell whether these equivalent ways of looking at exceptionality will help in the AI researcher's quest search for computationally tractable logics that cope with exceptions in a graceful way, though we do explore some ideas in this direction in Lecture 19. It would be interesting to reinterpret various nonmonotonic logics as logics of changing channels, but we have not attempted that here. We do feel that it can only help, though, to have an account of distributed systems and their information flow that can, in a principled way, say something sensible about reliability and exceptionality.

2.5 State Spaces

In science, engineering, and applied mathematics the classifications employed are typically derived from state-space models of a system. In the final section of this lecture we want to discuss how these models fit into the framework presented here. In particular, we want to show how any such state space gives rise to a local logic embodying a theory of information flow on the components of the system.

In the framework presented here, a state space S is not just a (possibly structured) set Ω_S of states, but also comes with a function mapping a set S of tokens into the set Ω_S of states. Such a state space S is depicted here as in the diagram below. (When we use the arrow notation and write *state*: $S \to \Omega_S$ horizontally, or vertically as below, we mean that *state* is a function with domain S and range contained in Ω_S.)



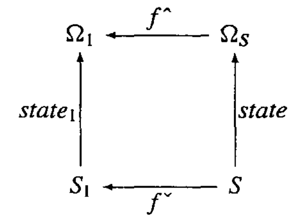
State Spaces and Classifications

Every state space is a classification, but this fact is a bit misleading. From an informational point of view, the types associated with a state space are not the states, but rather the events, that is, the sets $\alpha \subseteq \Omega_S$ of states. Thus for any state space S, we let Evt(S) be the classification with the same tokens as S but with types the sets $\alpha \in \text{pow}\,\Omega_S$ (the power set of Ω_S), and with $s \vDash \alpha$ if and only if $state(s) \in \alpha$. Notice that a token is of type $\alpha \cap \beta$ if and only if it is of both types α and β. More generally, using sets of states as types gives rise to a Boolean algebra, where the set-theoretic operations of intersection, union, and complement correspond to conjunction, disjunction, and negation, respectively.

State Spaces and Infomorphisms

Let us now turn to infomorphisms. Suppose we use the earlier state-space analysis of the system that consists of the roll of two dice. Let S consist of all tosses $\langle d_1, d_2, t\rangle$ where $d_1, d_2 \in D$ are distinct and Ω_S is the set used earlier, namely pairs of numbers between 1 and 6; let *state* be a function from S to Ω_S. Now let S_1 consist of all tosses $\langle d_1, t\rangle$ of one die, let Ω_1 be $\{1, 2, 3, 4, 5, 6\}$, and let $state_1$ be a function from S_1 to Ω_1.

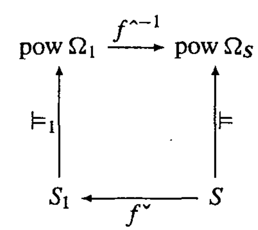
If these two state spaces are to sensibly describe a system and one of its parts, they should be related. How? Well, suppose we start with a toss $x = \langle d_1, d_2, t\rangle$ of two dice, extract a toss $x_1 = \langle d_1, t\rangle$ of the first dice, and determine its state $state_1(x_1)$. We should get the same thing if we determine the state $\langle n, m\rangle = state(x)$ of x and then take its first coordinate n. That is, writing $f^{\vee}(\langle d_1, d_2, t\rangle) = \langle d_1, t\rangle$ and $f^{\wedge}(\langle n, m\rangle) = n$, the following diagram should commute:



A pair $f = \langle f^{\wedge}, f^{\vee}\rangle$ like this is called a *projection* from the state space S for the whole to that for the component S_1. (In this diagram the state space for the system as a whole is written on the right and that for the component is written on the left, for ease of comparison with the notion of infomorphism.)

More generally, suppose one has a state-space analysis of some distributed system, and in addition, a state-space analysis of a component of the system. If both of these are correct, then there should be a way to get from the system to its component, and from the state of the system to the state of the component, in such a way that the above diagram commutes. That is, starting with c (in the lower right-hand corner), one could either determine its state and then see what information that gives one about the state of the component a_1, or one could determine the component and determine its state. These two procedures correspond to the two paths from the lower right to the upper left corner of the diagram. Clearly, the two ways of getting at the state of the component should agree.

The diagram makes it appear there is a problem squaring the notion of state-space projection with the notion of an infomorphism given earlier, because the arrows here both run from right to left rather than in opposite directions. As we have seen, however, when working with information, one works not with single states but with sets of states. The commutativity of the above diagram is equivalent to that of the following:



The function on types is now running the other direction, mapping any set α of states to its inverse image

$$f^{\wedge -1}[\alpha] = \{\sigma \mid f^{\wedge}(\sigma) \in \alpha\}.$$

Thus one has an infomorphism after all.

In Lecture 1 we mentioned the problem of squaring scientific accounts of inference and information flow with the more commonsense accounts actually used by people in going about the world. The use of infomorphisms and classifications also gives us a way to attack this problem by relating commonsense accounts of information flow to the kinds of state-space accounts more common in science and engineering. The basic idea is to obtain an infomorphism $f : A \rightleftarrows \mathrm{Evt}(S)$ from a commonsense classification A to the event classification associated with a state space S. An example of this is worked out in Lecture 3 (and in more detail in Lecture 18) where we show how useful, commonsense, but vague properties can be related to precise, scientific accounts.

State Spaces and Theories

The choice of a given set Ω_S as a set of states for a system implicitly brings with it a theory of that system. Consider, for example, the set Ω_S used in Example 2.3 for classifying throws $x = \langle d_1, d_2, t\rangle$ of a pair of dice. Suppose, for example, that die d_1 is normal but d_2 is abnormal in that it has a face with the numeral 7 (or seven dots) on it. In this case our state space is inadequate to model a toss $\langle d_1, d_2, t\rangle$ because the state of such a toss might not be in our state space. By assuming the set $\{\langle n, m\rangle \mid 1 \leq n, m \leq 6\}$ as our state space, we have implicitly limited ourselves to the analysis of particulars whose state is in this set. If we use this space in the presence of our unusual die, we will make unsound inferences.

On the other hand, imagine that the dice are six-sided dice but with only the numerals 1, 2, 3, and 4 on their faces, some more than once, of course. In this case our state-space model is inappropriate but in a different way. It implicitly assumes that every state $\sigma \in \Omega_S$ is possible, which in this case is not so. Reasoning using such a state space about these tosses will be incomplete. For example, from the information that the state of toss x has a sum of at least seven, it will fail to follow that the state is in $\{\langle 3, 4\rangle, \langle 4, 3\rangle\}$, an inference that is in fact warranted in this context.

Definition. For any state space S, $\mathrm{Log}(S)$ is the local logic on the classification Evt (S) with every token normal and with constraints given by

$$\Gamma \vdash \Delta \quad \text{iff} \quad \bigcap \Gamma \subseteq \bigcup \Delta$$

That is, $\Gamma \vdash \Delta$ holds if and only if every state that is in every type in Γ is in at least one type in Δ. Notice that this logic is sound, because we assume that every token a in S has a state $\mathrm{state}(a) \in \Omega_S$. But also notice that the logic is given entirely by the set Ω_S of states; it is completely independent of the set of tokens and the state function. This is quite different than the case of the sound and complete logic associated with a classification that depends crucially on the set of tokens and the classification relation.

Suppose we have an information channel with a core of the form $\mathrm{Evt}(S)$ for some state space S. The logic $\mathrm{Log}(S)$ is then a logic on this core that is suitable for distributing over the sum of the component classifications. This gives us an account of the information flow in the system implicit in the use of the set Ω_S of states of S. We will examine an application of this idea in the next lecture.

Let us now turn to an extended example to illustrate how these ideas can be put to work.

Lecture 3

A Simple Distributed System

In this lecture we investigate a distributed, physical system that is simple enough to explore in some detail but complex enough to illustrate some of the main points of the theory. Readers may work through this lecture as a way of getting a feeling for the motivations behind the mathematics that follow or skip ahead to the theory and come back to this example later. Readers who decide to work through this lecture should accept various assertions on faith, because they are justified by the work that follows.

Informal Description of the Circuit

The example consists of a light circuit LC. The circuit we have in mind is drawn from the home of one of the authors. It consists of a light bulb B connected to two switches, call them SW_1 and SW_2, one upstairs, the other downstairs. The downstairs switch is a simple toggle switch. If the bulb is on, then flipping switch SW_2 turns it off; if it is off, then flipping SW_2 turns it on. The upstairs switch SW_1 is like SW_2 except that it has a slider SL controlling the brightness of the bulb. Full up and the bulb is at maximum brightness (if lit at all); full down and the bulb is at twenty-five percent brightness, if lit, with the change in brightness being linear in between.

We are interested in showing how the theory sketched in the previous lecture and developed in this book handles two kinds of information flow in this system: static and dynamic. The static information we are interested in can be expressed informally by means of statements like the following:

1. If SW_1 is down and SW_2 is up, then B is on.
2. If SL is up and B is on, then B is bright.
3. If B is off, then the switches are both up or both down.

Another phrasing makes the connection with information flow clearer:

1. SW_1 being down and SW_2 being up carry the information that B is on.
2. SL being up and B being on carry the information that B is bright.
3. B being off carries the information that the switches are both up or both down.

Our aim is to see how these sorts of claims can be seen to fall out of an analysis of the circuit connecting the components B, SW_1, SW_2, and SL.

These claims are static because they do not involve any change in the system. The dynamic information that interests us has to do with changes effected by actions taken when someone uses the system by flipping one of the switches or moving the slider to a new position. We want to account for the following sorts of statements:

4. If B is off and SW_1 is flipped, then B will go on.
5. If B is off and SL is slid to the midpoint, then no change will take place in B.
6. If B is dim and SL is slid to the top, then B will be bright.

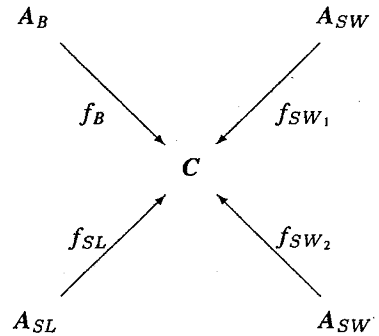
The aim of this lecture is to build an information channel $\mathcal{C}_{LC}$ and an associated local logic $\mathfrak{L}_{LC}$ that captures these regularities. In fact, we will build two such pairs, one for the static case and one for the dynamic. We call the static versions $\mathcal{C}_s$ and $\mathfrak{L}_s$ and the dynamic versions $\mathcal{C}_d$ and $\mathfrak{L}_d$. The most natural way to study the circuit LC is with state spaces. However, the regularities are phrased in terms of the commonsense conditionals used for talking about such circuits. Thus one of our tasks in analyzing this example is to show how these commonsense conditionals square with a state-space description of the system. The account will explain why these conditionals hold.

Notice, however, that these conditionals are defeasible. If the power to this house is off, then neither (2) nor (3) hold, for example. We want to see how this sort of thing fits into a general account. That is, normal and exceptional cases should be seen as arising from principled considerations involving the circuit. Indeed, they should be seen as following from the mathematical modeling of the principles enunciated in the preceding lecture.

As an added attraction, we note that our constraints involve useful but vague predicates like "dim" and "bright." Working out this example will allow us to see how such predicates can interact with our scientific description of the system by way of infomorphisms.[1]

[1] The basic idea here is elaborated in Lecture 18 on vagueness.

Our first task is to construct the classification A on which the local logic $\mathfrak{L}_s$ lives. This classification is the sum of four classifications, one for each of the components; $A = A_B + A_{SW} + A_{SW} + A_{SL}$. These classifications are connected together by a channel as follows:



A light bulb classification A_B. There are many different ways one could classify light bulbs at times. We could take into account the state of their vacuums, their filaments, the glass, their metal connectors, the current passing through them, and so on. We will, for now, at least, restrict ourselves to the illumination.

The tokens $b, b', \ldots$ of A_B consist of light bulbs at various times. Types consist of ON, OFF, BRIGHT, MEDIUM, and DIM, with the obvious classification relation. We assume that these types have the obvious meanings but stress that the meanings of the last three are vague and somewhat context dependent. We will see how that works out below.

The switch classification A_{SW}. Just as there are many ways of classifying the bulbs, so are there many ways of classifying the switches. We will keep things simple. The tokens $s, s', \ldots$ of S_{SW} consist of switches at various times. The types consist of UP and DN.

The slider classification A_{SL}. The tokens $sl, sl', \ldots$ of S_{SL} consist of sliders at various times. The types consist of real numbers between 0 and 1, representing the position of the slider, plus two new types UP and DN. The real-number types represent the position of the slider. For example, if $sl \models_{A_{SL}} .2$ then sl is exactly two-tenths of the way up. As for the other two types, we declare that sl is of type DN if and only if sl is less than one-tenth of the way up, and is of type UP if it is more than nine-tenths of the way up.

We now define the classification A we are after to be $A_B + A_{SW} + A_{SW} + A_{SL}$. The classification A_{SW} of switches occurs twice because the circuit we are interested in has two switches. The tokens of this classification consist of

4-tuples $\langle b, s_1, s_2, sl\rangle$, where b, s_1, s_2, and sl are instances of a bulb, two switches, and a slider SL. The types of this classification are the disjoint union of the types of the individual classifications. Because A_{SW} occurs twice, and because its types are also types in A_{SL}, the types of $A_B + A_{SW} + A_{SW} + A_{SL}$ contain three copies of UP and DN, one for each copy of A_{SW} and one for A_{SL}. (The other types are distinct from one another.) We denote the copies of UP by UP_1, UP_2, and UP_{SL}, respectively, and analogously define copies for DN.

We want a local logic on A that captures the intuitions about information flow in the system that interests us. Intuitively, this local logic should have as normal tokens those 4-tuples $\langle b, s_1, s_2, sl\rangle$ that are part of a normal instance of our circuit, and it should have as constraints things like

$$\text{DN}_1, \text{UP}_2 \vdash \text{ON}.$$

$$\text{UP}_{SL}, \text{ON} \vdash \text{BRIGHT}.$$

We will obtain such a local logic by finding a natural channel C_s with these four classifications as its component classifications and by looking at a distributed local logic associated with this system.

The most natural way to proceed for this particular example is by way of state spaces. So we first construct state spaces for each of these components together with infomorphisms of these classifications into the event classifications of the state spaces. Because our analysis will make central use of state spaces, we need to say a bit more about them and their relationship to classifications.

First, recall that a state space S consists of a set S of tokens, a set Ω of states, and a function $state : S \to \Omega$ assigning a state to each token. We call S *complete* if every state $\sigma \in \Omega$ is the state of some $s \in S$.

Associated with any state space S, there is a classification Evt(S), called the *event classification of S*. The tokens of Evt(S) are the same as those of S but whose types are sets of states from S, interpreted disjunctively. That is, if α is a set of states of S and a is a token of S, then $a \models_{\text{Evt}(S)} \alpha$ if and only if the state(a), the state of a, is in α. As we mentioned, in this classification, the types form a Boolean algebra under the operations of intersection (conjunction of types), union (disjunction of types), and complement (negation of types).

There is an intuitive notion of entailment implicit in any classification of the form Evt(S). Recall that type α entails type β if every state in α is in β; that is, if $\alpha \subseteq \beta$. When we combine this with the Boolean operations, we see that, for example, $\alpha_1, \alpha_2 \vdash \beta_1, \beta_2$ if every state in both α_1 and α_2 is in at least one of β_1 or β_2, that is, if $\alpha_1 \cap \alpha_2 \subseteq \beta_1 \cup \beta_2$. More generally, if Γ and Δ are sets of Evt(S) types, then, intuitively,

$$\Gamma \vdash \Delta \quad \text{iff} \quad \bigcap \Gamma \subseteq \bigcup \Delta.$$

This defines a sound entailment relation on Evt(S). We can turn it into a sound local logic by simply taking all the tokens of S to be normal. This logic is called the *local logic induced by the state space* S, and is written Log(S); it is the logic implicit in the use of the state-space model S.

We can now see the rationale for calling a space S complete when every state is the state of some token. For suppose S is complete and consider a sequent $\langle \Gamma, \Delta \rangle$ that holds of every token. It follows that $\Gamma \vdash \Delta$, for if not, there would be a state in $\sigma \in (\bigcap \Gamma - \bigcup \Delta)$. But then let s be a token whose state is σ. This token would be of every type in Γ and of no type in Δ, contradicting the assumption that $\langle \Gamma, \Delta \rangle$ holds of every token. In other words, the logic Log(S) is a complete logic.

In the previous lecture we noted that whenever there is a projection $p : S \rightrightarrows S_1$ of state spaces, there is a natural associated infomorphism p^*: Evt(S_1) $\rightleftarrows$ Evt(S) of their associated classifications. On types, this infomorphism takes a set of types to its inverse image under $p^{\wedge}$; on tokens it is identical to $p^{\vee}$. We call this infomorphism Evt(p). Thus whenever p is a projection, Evt(p) : Evt(S_1) $\rightleftarrows$ Evt(S) is an infomorphism. We will use this operation repeatedly below.

3.1 The Static Case

We begin by restricting attention to the static case, returning to the dynamic case in the next section, where we will build on the work done here.

The bulb state space S_B. The tokens $b, b', \ldots$ of S_B consist of light bulbs at various times. The states will consist of real numbers between 0 and 1, representing the brightness of the bulb. If $\text{state}_{S_B}(b) = .5$ then b is at half brightness; if $\text{state}_{S_B}(b) = 0$ then b is off. The classification Evt(S_B) has the same tokens as A_B and has types arbitrary subsets of the closed, unit interval [0, 1].

Infomorphisms from A_B to Evt(S_B). The classification A_B is rather subjective. What is considered bright or dim varies from person to person and also with conditions. What seems bright at night may not seem bright at noon on a sunny day. This sort of dependence on the viewer's perspective might seem inimical developing a mathematical theory of information flow, so we want to see how we can handle it.

One way to get at some aspects of this subjectivity is by recognizing that there are many different infomorphisms $g_B : A_B \rightleftarrows$ Evt(S_B) from A_B into Evt(S_B).

We are interested in those that are the identity on tokens and satisfy the following condition on types:

- g_B(ON) = (0, 1]
- g_B(OFF) = {0}
- g_B(DIM) is a left closed subset of (0, 1].[2]
- g_B(BRIGHT) is a right closed subset of (0, 1].
- g_B(MEDIUM) is a convex subset of (0, 1].[3]
- Every member of g_B(DIM) is less than every member of g_B(MEDIUM), which in turn is less than every member of g_B(BRIGHT).

Notice that we have not assumed that the three sets g_B(MEDIUM), g_B(DIM), and g_B(BRIGHT) exhaust the interval (0, 1], because there might be some values that the observer would be reluctant to classify as any of medium, dim, or bright.

The switch state space S_{SW}. The tokens $s, s', \ldots$ of S_{SW} consist of switches at various times. The states will consist of 0 and 1. The state of s is 1 if s is up, 0 if s is down.

Whenever a state space S has exactly two states, say 0 and 1, the event space Evt(S) will have four types, namely, the empty set $\emptyset$, the set {0, 1}, and the types {0} and {1}. The empty set is a type with no tokens of that type; it represents impossibility. The type {0, 1} represents no information at all, because every token is of that type; it represents necessity. So the only possible but nonnecessary types are {0} and {1}. Thus we see that for such a state space, the only advantage of Evt(S) over S is that the former has an impossible type and a necessary type.

We define a token-identical infomorphism $g_{SW} : A_{SW} \rightleftarrows \text{Evt}(S)_{SW}$ to capture the relationship between the types and the states. Namely, g_{SW}(UP) = {1} and g_{SW}(DN) = {0}.

The slider state space S_{SL}. The tokens $sl, sl', \ldots$ of S_{SL} consist of sliders at various times. The states will consist of real numbers between 0 and 1, representing the position of the slider. If $\text{state}_{S_{SW}}(sl) = .2$ then sl is two-tenths of the way up. The classification Evt(S_{SL}) has arbitrary subsets of [0, 1] as types.

[2] A left closed subset X of a set Y of reals is a set such that if $x \in X$ and $y \in Y$ is less than x, then $y \in X$.

[3] A convex subset X of a set Y of reals is a set such that if $x_1, x_2 \in X$, and $y \in Y$ is between x_1 and x_2, then $y \in X$.

We define the obvious infomorphism $g_{SL}: A_{SL} \rightleftarrows \mathrm{Evt}(S_{SL})$ by letting $g_{SL}(r) = \{r\}$ for each real number and letting $g_{SL}(\mathrm{DN}) = [0, .1)$ and $g_{SL}(\mathrm{UP}) = (.9, 1]$.

We have now constructed token identical infomorphisms from each of our classifications into the event classifications of the corresponding state spaces.

The Channel C_s

The information channel consists of a classification of the circuit as a whole, together with infomorphisms from our component classifications into this classification. In coming up with a classification for the circuit, we have to choose what aspects of the circuit to model. Should we take into account the condition of the wiring, the amount of current available, whether or not the house is flooded, or just what? We want our framework to be able to account for such factors and contingencies without getting bogged down by them. We choose to develop a state space S_L to model the system and then take the core of the channel to be the event classification of this state space. Before defining S_L, we define an auxiliary space S_L^*.

The state spaces S_L^*. The tokens $c, c', \ldots$ of S_L^* are arbitrary instances of the circuit at various times. The set Ω^* of states of S_L^* is $[0, 1]^4$, the set of all 4-tuples of real numbers between 0 and 1. This is a very simplified model of the state of the circuit. The circuit c is in state $\langle r_1, r_2, r_3, r_4\rangle$ if the state of c's bulb is r_1, the state of c's lower switch is r_2, the state of c's upper switch is r_3, and the state of c's slider is r_4.

We have set up our example state spaces so that there are simple, natural projections $p_B : S_L^* \rightrightarrows S_B$, $p_{SW1} : S_L^* \rightrightarrows S_{SW}$, $p_{SW2} : S_L^* \rightrightarrows S_{SW}$, and $p_{SL} : S_L^* \rightrightarrows S_{SL}$ from S_L^* into the state spaces for the bulbs, switches, and slider, respectively.

For example, the projection $p_B : S_L^* \rightrightarrows S_B$ acts as follows. On tokens, $p_B(c)$ is the bulb that is part of the circuit instance c. (We suppose that every circuit instance that is a token of S_L^* has a bulb screwed in.) On states, $p_B(\langle r_1, r_2, r_3, r_4\rangle) = r_1$. The projection $p_{SW_1} : S_L^* \rightrightarrows S_{SW}$ acts on tokens by $p_{SW1}(c)$ being the instance of the switch SW_1 that is part of the circuit instance c. On states, $p_{SW_1}(\langle r_1, r_2, r_3, r_4\rangle) = r_2$. The projection $p_{SW_2} : S_L^* \rightrightarrows S_{SW}$ is similar except that it gives an instance of SW_2 on tokens and r_3 on states. The projection $p_{SL} : S_L^* \rightrightarrows S_{SL}$ is defined similarly.

The state space S_L. The constraints we are after are a product of natural laws governing the circuit and the meanings of our types. The latter are built into the infomorphisms defined above. The laws governing the circuit, however, have not yet been built into the model. We do this by looking at a subspace S_L of S_L^*.

Let Ω consist of all those tuples $\sigma = \langle r_1, r_2, r_3, r_4\rangle$ satisfying:

$$r_2, r_3 \in \{0, 1\}$$

$$r_1 = |r_2 - r_3| \left(\frac{3r_4 + 1}{4}\right)$$

Notice that if $r_2 = r_3$ (that is, if the switches are both up or both down), then $r_1 = 0$. On the other hand, if $r_2 = 1$ and $r_3 = 0$ or vice versa (meaning that the two switches are in opposite positions), then

$$r_1 = \frac{3r_4 + 1}{4}.$$

For example, if $r_4 = 0$, then $r_1 = .25$, meaning that if the slider is all the way down, then the bulb is at twenty-five percent brightness, whereas if $r_4 = 1$, then $r_1 = 1$, meaning that if the slider is all the way up, then the bulb is at full brightness. The dependence of r_1 on r_4 is linear, as specified earlier.

We let S_L be the subspace of S_L^* with Ω as the set of states and tokens those instances of the circuit whose state is in Ω. It is reasonable to suppose that this state space is complete for the following reason. There are only four possible combinations of positions for the switches. We have put the switches in these four positions and, in each of these, run the slider from bottom to top. In this way we have exhausted all the possible states of the system, at least up to the gap between the physical system and its mathematical model.

The core of the channel. The core of our channel C_s is the event classification of S_L, that is, the classification $C = \mathrm{Evt}(S_L)$.

The infomorphisms from the component classifications are defined in the expected manner:

The infomorphism $f_B : A_B \rightleftarrows C$ is the composition of the infomorphisms $g_B : A_B \rightleftarrows \mathrm{Evt}(S_B)$ and $\mathrm{Evt}(p_B) : \mathrm{Evt}(S_B) \rightleftarrows C$.[4]
The infomorphism $f_{SW_1} : A_{SW} \rightleftarrows C$ is the composition of $g_{SW} : A_{SW} \rightleftarrows \mathrm{Evt}(S_{SW})$ and $\mathrm{Evt}(p_{SW_1}) : \mathrm{Evt}(S_{SW}) \rightleftarrows C$.
The infomorphism $f_{SW_2} : A_{SW} \rightleftarrows C$ is the composition of $g_{SW} : A_{SW} \rightleftarrows \mathrm{Evt}(S_{SW})$ and $\mathrm{Evt}(p_{SW_2}) : \mathrm{Evt}(S_{SW}) \rightleftarrows C$.
The infomorphism $f_{SL} : (A_{SL}) \rightleftarrows C$ is the composition of $g_{SL} : A_{SL} \rightleftarrows \mathrm{Evt}(S_{SL})$ and $\mathrm{Evt}(p_{SL}) : \mathrm{Evt}p(S_{SL}) \rightleftarrows C$.

[4] Composition of infomorphisms is defined in the obvious manner in the next lecture.

Let the index set $I = \{B, SW_1, SW_2, SL\}$, and let

$$\mathcal{C}_s = \{f_i : \boldsymbol{A}_i \rightleftarrows \boldsymbol{C}\}_{i \in I}$$

be the channel as depicted earlier.

The core classification $\boldsymbol{C}$ has a local logic $\mathfrak{L}_C$ that captures the state space classification, namely, the logic Log($\boldsymbol{S}_L$) defined earlier. This local logic has a consequence relation given by

$$\Gamma \vdash \Delta \quad \text{iff} \quad \bigcap \Gamma \subseteq \bigcup \Delta$$

for Γ, Δ sets of sets of states of Ω. It is sound, so its normal tokens are the tokens of $\boldsymbol{S}_L$, that is, the tokens of $\boldsymbol{S}_L^*$ whose state obey the defining law for Ω. Because the state space is complete, the local logic is also complete.

The local logic $\mathfrak{L}_s$. We can now use the local logic $\mathfrak{L}_C$ on the core $\boldsymbol{C}$ to obtain the desired local logic $\mathfrak{L}_s$ on the sum $\boldsymbol{A}$ using the infomorphisms. Let $f : \boldsymbol{A} \rightleftarrows \boldsymbol{C}$ be the sum infomorphism $\sum_{i \in I} f_i$. This infomorphism allows us to move the logic from $\boldsymbol{C}$ to $\boldsymbol{A}$, via f-Elim. $\mathfrak{L}_s$ is the resulting local logic. In the lectures that follow, this kind of logic will be denoted by the logic $f^{-1}[\mathfrak{L}_C]$. It is the strongest logic on $\boldsymbol{A}$ such that the infomorphism f is well-behaved. In particular, $\Gamma \vdash_{\mathfrak{L}_s} \Delta$ if and only if $f[\Gamma] \vdash_{\mathfrak{L}_C} f[\Delta]$. (We'll compute some of the constraints in a moment.) This local logic is complete, because the inverse image of a complete local logic is always complete. It is not sound, however. That is, not all tokens are normal. The normal tokens consist of those sequences $\langle b, s_1, s_2, sl\rangle$ of bulb, switches, and slider that are connected by means of a circuit c whose state satisfies the defining law for Ω.

Because the local logic $\mathfrak{L}_s$ is complete, it should give us all the constraints we expect, but let us check the following two constraints in some detail, by way of illustrating this logic:

$$\text{DN}_1, \text{UP}_2 \vdash_{\mathfrak{L}_s} \text{ON},$$

and

$$\text{UP}_{SL}, \text{ON} \vdash_{\mathfrak{L}_s} \text{BRIGHT}.$$

By the definition of $\mathfrak{L}_s$, we need to check that

$$f_{SW_1}(\text{DN}_1), f_{SW_2}(\text{UP}_2) \vdash_{\mathfrak{L}_C} f_B(\text{ON}),$$

and

$$f_{SL}(\text{UP}_{SL}), f_B(\text{ON}) \vdash_{\mathfrak{L}_C} f_B(\text{BRIGHT}).$$

These amount to

$$f_{SW_1}(\text{DN}_1) \cap f_{SW_2}(\text{UP}_2) \subseteq f_B(\text{ON}),$$

and

$$f_{SL}(\text{UP}_{SL}) \cap f_B(\text{ON}) \subseteq f_B(\text{BRIGHT}).$$

These, in turn, say the following: for every for tuple $\langle r_1, r_2, r_3, r_4\rangle \in [0, 1]^4$ such that $r_2, r_3 \in \{0, 1\}$ and

$$r_1 = |r_2 - r_3|\left(\frac{3r_4 + 1}{4}\right)$$

1. if $r_2 = 0$ and $r_3 = 1$, then $r_1 > 0$;
2. if $r_4 > .9$ and $r_1 > 0$, then $r_1 \in f_B(\text{BRIGHT})$.

The first of these is clear. The second does not actually follow from anything we have said so far. The reason is that in postulating our conditions on the infomorphism g_B we did not assume that $g_B(\text{BRIGHT})$ is nonempty; it might be, for example, that a person might under certain conditions consider the bulb dim even at its brightest. But for such a person, the second constraint would not hold! To get this constraint to hold, we must assume that $g_B(\text{BRIGHT})$ is nonempty. In fact, because anything over nine-tenths of the way up counts as being up, this constraint will hold if and only if we assume that all brightnesses above 92.5% count as being bright, that is, that $(.925, 1] \subseteq f_B(\text{BRIGHT})$ (the reason being that $.925 = (3 \times .9 + 1)/4$).

Let us now consider the third constraint

If B is off, then the switches are both up or both down.

Intuitively, this should be expressed by

$$\text{OFF} \vdash_{\mathfrak{L}_s} \text{UP}_1 \wedge \text{UP}_2, \text{DN}_1 \wedge \text{DN}_2.$$

However, our local logic $\mathfrak{L}_s$ does not have a conjunction because $\boldsymbol{A}$ doesn't have one, so this does not make sense in this logic.

The natural thing to do is to enlarge the system of types in $\boldsymbol{A}$ to one that has conjunctions, disjunctions, and negations of types. There is a canonical way to do this, resulting in a classification Boole($\boldsymbol{A}$). Because the types of $\boldsymbol{C}$ form a complete Boolean algebra, there is a canonical way to extend the infomorphism f to an infomorphism $f^* = \text{Boole}(f)$ from Boole($\boldsymbol{A}$) to $\boldsymbol{C}$. Let $\mathfrak{L}_s^*$ be the local logic $f^{*-1}[\mathfrak{L}_C]$. This is the natural extension of our logic $\mathfrak{L}_s$ to the Boolean closure Boole(A) of A. We will show that

$$\text{OFF} \vdash_{\mathfrak{L}_s^*} \text{UP}_1 \wedge \text{UP}_2,\ \text{DN}_1 \wedge \text{DN}_2.$$

This is equivalent to the condition:

$$f_B(\text{OFF}) \subseteq (f_{SW_1}(\text{UP}_1) \cap f_{SW_2}(\text{UP}_2)) \cup (f_{SW_1}(\text{DN}_1) \cap f_{SW_2}(\text{DN}_2))$$

This says the following: for every 4-tuple $\langle r_1, r_2, r_3, r_4 \rangle \in [0, 1]^4$ such that $r_2, r_3 \in \{0, 1\}$ and

$$r_1 = |r_2 - r_3| \left(\frac{3r_4 + 1}{4} \right),$$

if $r_1 = 0$, then either $r_2 = 1$ and $r_3 = 1$ or $r_2 = 0$ and $r_3 = 0$. This is easily checked.

Exceptions

Our model of the circuit made many simplifying assumptions of normality, assumptions that get reflected in our constraints. Let us see what happens if we relax one of these assumptions. Suppose, for example, that we want to model the possibility that the power to the house can go off, during a storm, say. To do this, construct a new state space S'_L using 5-tuples rather than 4-tuples for states, say $\langle r_1, \ldots, r_5 \rangle \in [0, 1]^5$, where the fifth coordinate represents whether there is power ($r_5 = 1$) or not ($r_5 = 0$).

Let Ω' consist of all those tuples $\sigma = \langle r_1, r_2, r_3, r_4, r_5 \rangle$ satisfying

$$r_2, r_3, r_5 \in \{0, 1\}$$

$$r_1 = |r_2 - r_3| \left(\frac{3r_4 + 1}{4} \right) r_5$$

and let S'_L be the space with this set of states and tokens that are all circuits whose state is in Ω'. This will properly include the tokens of our earlier example, which only included circuits where the power is on. Similarly, let $C' = \text{Evt}(S'_L)$ be the event classification of S'_L.

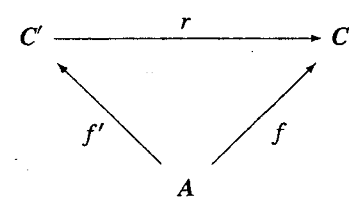
We could run through everything we did before and obtain a channel

$$\mathcal{C}'_s = \{f'_i : A_i \rightleftarrows C'\}_{i \in I}.$$

Using this local logic would not give us any of the above constraints, because this channel takes into account the possibility of the power being off. To get some constraints, we would need to devise a classification A_P for the power and put it in as a summand of our classification A and put the assumption that the power was on into the antecedent of the constraint.

The channel $\mathcal{C}'_s$ is a refinement of the channel $\mathcal{C}_s$ in the precise sense defined in the preceding lecture. This means that its constraints are more reliable than those of $\mathcal{C}_s$, that is, more reliable in that they hold of a wider class of cases. To see that $\mathcal{C}'_s$ is a refinement of $\mathcal{C}_s$, first notice that the state space S_L is isomorphic to a subspace of the state space S'_L; the embedding e is the identity on tokens whereas on types $e(\langle r_1, r_2, r_3, r_4 \rangle) = \langle r_1, r_2, r_3, r_4, 1 \rangle$.

Applying this operation to e, we let $r = \text{Evt}(e)$ be the infomorphism that is the identity on tokens and is defined on types by: $r(\alpha) = e^{-1}[\alpha]$. The infomorphism r shows the channel $\mathcal{C}'_s$ to be a refinement of the channel $\mathcal{C}_s$.



The constraints discussed above hold relative to the channel $\mathcal{C}_s$ but not relative to the more refined channel $\mathcal{C}'_s$.

Suppose that one has the information that the light is off and the switch SW_2 is up. Does one have the information that the switch SW_1 is also up? Well, not unless the two are connected by a normal instance c of our circuit, of course, and so are both up at the same time. But even then we see that the constraint depends upon which channel we consider. If we consider the channel $\mathcal{C}_s$, then the answer is "yes," because

$$\text{OFF}, \text{UP}_2 \vdash_{\mathfrak{L}_s} \text{UP}_1$$

and because we are dealing with a normal token of the local logic. However, if we are considering the more refined channel $\mathcal{C}'_s$, then the answer is "no," because

$$\text{OFF}, \text{UP}_2 \nvdash_{\mathfrak{L}'_s} \text{UP}_1.$$

For example, the state $\sigma = \langle 0, 1, 0, 1, 0 \rangle$ is a counterexample, one where the power is off.

Our general attitude toward defeasible reasoning and its so-called nonmonotonicity is that additional information can alert the reasoner to a shift in channels and so to a shift in local logics. This happens typically by bringing in some issue that is not relevant to the channel under consideration. Thus, for example, suppose one is asked "Does the above constraint hold when the power is off?" In such a case one knows immediately that channel C_s and its associated local logic $\mathfrak{L}_s$ are inappropriate and moves to the local logic $\mathfrak{L}'_s$.

3.2 The Dynamic Case

Recall that we are also concerned with information that involves someone operating the switches and slider, things like the following:

4. If B is off and SW_1 is flipped, then B will go on.
5. If B is off and SL is slid to the midpoint, then no change will take place in B.
6. If B is dim and SL is slid to the top, then B will be bright.

We now want to find a classification $\boldsymbol{B}$ and a local logic $\mathfrak{L}_d$ on $\boldsymbol{B}$ that captures these kinds of constraints.

The constraints are about actions that affect the circuit so we need to classify such actions. We thus define a classification $\boldsymbol{Act}$ of actions. The tokens are particular acts $a, a', \ldots$, that involve our circuit. We assume that for each act a there are two instances of the circuit, init(a) and final(a). We posit types FLIP1, FLIP2, RAISE-SL, SLIDER-TO-p (where $0 \leq p \leq 1$), and TURN-OFF with the natural extensions. For example, the extension of FLIP1 is the set of all acts where the first switch is flipped, but the other switch and slider are left unchanged. Similarly, the extension of SLIDER-TO-p is the set of all acts where the slider starts at some position other then p and is positioned at p at the end, with both switches remaining changed.

Our desired classification is the classification $\boldsymbol{B} = \boldsymbol{A} + \boldsymbol{Act} + \boldsymbol{A}$. The tokens consist of 9-tuples of the form

$$\langle b^{init}, s_1^{init}, s_2^{init}, sl^{init}, a, b^{final}, s_1^{final}, s_2^{final}, sl^{final} \rangle.$$

We interpret such a token as follows: The first four terms will be the initial components, before the action a; the last four will be the final components (after the action a). Because the classification has two copies of $\boldsymbol{A}$ it will have two copies of each of its types in the type set. We use superscripts to indicate these. For example, UP_1^{init} is the copy of UP_1 that goes with the first copy of $\boldsymbol{A}$, because that copy represents the initial configuration. Our example constraints can now be formulated.

4. OFF^{init}, FLIP1 $\vdash$ ON^{final}.

5. OFF^{init}, SLIDER-TO-.5 $\vdash$ OFF^{final}.
6. DIM^{init}, SLIDER-TO-1 $\vdash$ BRIGHT^{final}.

The next step in our analysis is to translate these types into types in the event space of an appropriate state space. For this we use a state space $\boldsymbol{S}_{Act}$. The tokens consist of all acts a such that the states of both init(a) and final(a) are in the set Ω of normal states discussed in the previous section. The set of states is $\Omega \times \Omega$. The state of an action is defined to be the ordered pair consisting of the states of its initial and final circuits.

Let $p_{init} : \boldsymbol{S}_{Act} \rightrightarrows \boldsymbol{S}_L$ and $p_{final} : \boldsymbol{S}_{Act} \rightrightarrows \boldsymbol{S}_L$ be the projections defined as follows:

$$\begin{aligned} p_{init}(a) &= \text{init}(a) \\ p_{init}(\langle \sigma, \sigma' \rangle) &= \sigma \\ p_{final}(a) &= \text{final}(a) \\ p_{final}(\langle \sigma, \sigma' \rangle) &= \sigma' \end{aligned}$$

Our information channel has as its core the event classification $\boldsymbol{C}_{Act} = \text{Evt}(\boldsymbol{S}_{Act})$. This classification has as types subsets of $\Omega \times \Omega$, that is, binary relations on Ω.

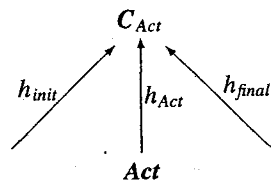
Because $\boldsymbol{C}_{Act}$ is the event space of the state space $\boldsymbol{S}_{Act}$, there is an inherited local logic, namely, $\text{Log}(\boldsymbol{S}_{Act})$. We are going to use this local logic to obtain the local logic $\mathfrak{L}_d$ on the classification $\boldsymbol{B}$ by means of an infomorphism $h : \boldsymbol{B} \rightleftarrows \boldsymbol{C}_{Act}$. We construct this infomorphism as follows. First, let $k_{init} = \text{Evt}(p_{init})$ and $k_{final} = \text{Evt}(p_{final})$ so that $k_{init} : \boldsymbol{C} \rightleftarrows \boldsymbol{C}_{Act}$ and $k_{final} : \boldsymbol{C} \rightleftarrows \boldsymbol{C}_{Act}$ are infomorphisms. We let $h_{init} = f k_{init}$ and $h_{final} = f k_{final}$, where $f : \boldsymbol{A} \rightleftarrows \boldsymbol{C}$ is the infomorphism used in the previous section. Thus both h_{init} and h_{final} are infomorphisms from $\boldsymbol{A}$ into $\boldsymbol{C}$. The first represents the case where the components are considered components of the initial circuit of an action; the latter represents the case where the components are considered components of the final circuit of the action.

There is a token-identical infomorphism $h_{Act} : \boldsymbol{Act} \rightleftarrows \boldsymbol{C}_{Act}$ defined on types α of $\boldsymbol{Act}$ as follows:

α	$h_{Act}(\alpha)$
FLIP1	$\{\langle \sigma, \sigma' \rangle \mid r'_{SW_1} = 1 - r_{SW_1}\}$
FLIP2	$\{\langle \sigma, \sigma' \rangle \mid r'_{SW_2} = 1 - r_{SW_2}\}$
RAISE-SL	$\{\langle \sigma, \sigma' \rangle \mid r'_{SL} > r_{SL}\}$
SLIDER-TO-p	$\{\langle \sigma, \sigma' \rangle \mid r_{SL} \neq p, r'_{SL} = p\}$
TURN-OFF	$\{\langle \sigma, \sigma' \rangle \mid r_B = 1, r'_B = 0\}$

(In the descriptions on the right we have only given the condition on the coordinate that changes under the action. All other coordinates are fixed.)

The local logic $\mathfrak{L}_d$. We are now in a position to define the channel C_d that is going to give us our desired local logic. Namely, it consists of the three infomorphisms displayed below:



Adding these three infomorphisms together, we obtain an infomorphism $h = h_{init} + h_{Act} + h_{final}$ from $\boldsymbol{B} = \boldsymbol{A} + \boldsymbol{Act} + \boldsymbol{A}$ into $\boldsymbol{C}_{Act}$. This infomorphism $h\colon \boldsymbol{B} \rightleftarrows \boldsymbol{C}_{Act}$ is the desired map for obtaining our local logic. Let $\mathfrak{L}_d = h^{-1}[\text{Log}(\boldsymbol{S}_{Act})]$, a local logic on the classification $\boldsymbol{C}_{Act}$.

The normal tokens of this local logic are those 9-tuples

$$\langle b^{init}, s_1^{init}, s_2^{init}, sl^{init}, a, b^{final}, s_1^{final}, s_2^{final}, sl^{final}\rangle$$

such that a is an action for which both init(a) and final(a) are normal circuit instances and such that the components of init(a) are $b^{init}, s_1^{init}, s_2^{init}, sl^{init}$ and the components of final(a) are $b^{final}, s_1^{final}, s_2^{final}, sl^{final}$.

Let us now check our three sample constraints, to see that they in fact hold in our local logic:

4. OFF^{init}, FLIP1 $\vdash_{\mathfrak{L}_d}$ ON^{final}.
5. OFF^{init}, SLIDER-TO-.5 $\vdash_{\mathfrak{L}_d}$ OFF^{final}.
6. DIM^{init}, SLIDER-TO-1 $\vdash_{\mathfrak{L}_d}$ BRIGHT^{final}.

Let $\langle r_B^{init}, r_{SW_1}^{init}, r_{SW_2}^{init}, r_{SL}^{init}\rangle, \langle r_B^{final}, r_{SW_1}^{final}, r_{SW_2}^{final}, r_{SL}^{final}\rangle \in [0,1]^4$ meet the conditions for states in Ω. The three constraints translate into the following conditions:

4. If $r_B^{init} = 0$ and $r_{SW_1}^{final} = 1 - r_{SW_1}^{init}$, then $r_B^{final} > 0$.
5. If $r_B^{init} = 0$ and $r_{SL}^{final} = .5$, then $r_B^{final} = 0$.
6. If $r_{SL}^{init} \in f_B(\text{DIM})$ and $r_{SL}^{final} = 1$, then $r_{SL}^{init} \in f_B(\text{BRIGHT})$.

The first two are readily checked. The last one can be checked if we again assume that $(.925, 1] \subseteq f_B(\text{BRIGHT})$.

Sequences of Actions

The logic $\mathfrak{L}_d$ handles single actions but it does not allow us to express constraints like the following:

7. If B is off and SW_1 is flipped and then SW_2 is flipped, then B will be off.
8. If B is off, SL is slid to the bottom, and then SW_2 is flipped, then B will be dim.

This shortcoming is easily remedied, however.

Assume there is an operation $\circ$ of *composition* of actions, so that if a_1, a_2 are actions with final(a_1) = init(a_2) then $a_1 \circ a_2$ is an action with init($a_1 \circ a_2$) = init(a_1) and final($a_1 \circ a_2$) = final(a_2). At the level of types, add a binary operation, also denoted by $\circ$, requiring that a is of type $\alpha_1 \circ \alpha_2$ if and only if there are actions a_1 of type α_1 and a_2 of type α_2 such that $a = a_1 \circ a_2$. Finally, we define $h_{Act}(\alpha_1 \circ \alpha_2)$ to the (relational) composition of the relations $h_{Act}(\alpha_1)$ and $h_{Act}(\alpha_2)$. With these elaborations the reader can easily derive the following formal versions of our earlier constraints:

7. OFF^{init}, FLIP1 $\circ$ FLIP2 $\vdash$ OFF^{final}.
8. OFF^{init}, SLIDER-TO-0 $\circ$ FLIP2 $\vdash$ DIM^{final}.

Part II

Channel Theory

Lecture 4

Classifications and Infomorphisms

In this lecture, we present the basic theory of classifications, infomorphisms, and information channels, the latter being our basic model of a distributed system. Information flow will be defined relative to an information channel. These notions are used throughout the rest of the book.

4.1 Classifications

Information presupposes a system of classification. Information is only possible where there is a systematic way of classifying some range of things as being this way or that.

Information flow is not restricted to a single on to logical category. The theoretical vocabularies used to describe these different kinds of things can be extremely diverse, so one needs a way to talk about classification that does not favor any particular "vocabulary." This is captured in the following definition.

Definition 4.1. A *classification* $A = \langle \text{tok}(A), \text{typ}(A), \models_A \rangle$ consists of

1. a set, tok(A), of objects to be classified, called the *tokens of* A,
2. a set, typ(A), of objects used to classify the tokens, called the *types of* A, and
3. a binary relation, $\models_A$, between tok(A) and typ(A).

If $a \models_A \alpha$, then a is said to be *of type* α *in* A. We sometimes represent a classification by means of a diagram of the following form:

$$\begin{array}{c} \text{typ}(A) \\ \Big| \models_A \\ \text{tok}(A) \end{array}$$

Example 4.2. Finite classifications can be conveniently represented by a *classification table.* We write the types along the top of the table, and tokens along the left side. For example, the following table

$A_{4.2}$	α_1	α_2	α_3	α_4	α_5
a_1	1	1	0	1	0
a_2	1	0	1	0	1
a_3	0	1	0	1	0
a_4	1	0	1	0	0
a_5	0	1	0	0	0
a_6	1	0	0	0	1

represents the classification A with tokens $\{a_1, a_2, a_3, a_4, a_5, a_6\}$ and types $\{\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5\}$, where a_1 is of type α_1, α_2, and α_4, whereas a_2 is of types α_1, α_3, and α_5. Notice that although there are five types, there are $2^5 = 32$ possible distinct sequences of 1s and 0s. Any classification where some of the possible rows do not appear in the table is one where there are patterns or regularities among the types. In the above classification, for example, we see that nothing that is of type α_3 is of type α_4 and that every token of type α_5 is also of type α_1.

Example 4.3. We can represent the classification of English words according to parts of speech as given in Webster's Dictionary using a classification table. We give only a small part of the table representing this classification, of course.

Webster	NOUN	INT VB	TR VB	ADJ
bet	1	1	1	0
eat	0	1	1	0
fit	1	1	1	1
friend	1	0	1	0
square	1	0	1	1
⋮				

Given a classification A, we introduce the following notation and terminology. For any token $a \in \mathrm{tok}(A)$, the *type set* of a is the set

$$\mathrm{typ}(a) = \{\alpha \in \mathrm{typ}(A) \mid a \models_A \alpha\}.$$

Similarly, for any type $\alpha \in \mathrm{typ}(A)$, the *token set* of α is the set

$$\mathrm{tok}(\alpha) = \{a \in \mathrm{tok}(A) \mid a \models_A \alpha\}.$$

(For example, in the classification **Webster**, the type set of the word "square" is the set {NOUN, TRANSITIVE VERB, ADJECTIVE}. The token set of the type NOUN is the set of all nouns of English.)

Types α_1 and $\alpha_2 \in \mathrm{typ}(A)$ are *coextensive in* A, written $\alpha_1 \sim_A \alpha_2$, if $\mathrm{tok}(\alpha_1) = \mathrm{tok}(\alpha_2)$. Tokens a_1, a_2 are *indistinguishable in* A, written $a_1 \sim_A a_2$, if $\mathrm{typ}(\alpha_1) = \mathrm{typ}(\alpha_2)$. (No pair of distinct types in Example 4.2 or in **Webster** are coextensive. In Example 4.2 it happens that no pair of tokens are indistinguishable. In **Webster** any two words that are of the same parts of speech, like "apple" and "kiwi" are indistinguishable.)

The classification A is *separated* provided there are no distinct indistinguishable tokens, that is, if $a_1 \sim_A a_2$ implies $a_1 = a_2$. The classification A is *extensional* provided that all coextensive types are identical, that is, if $\alpha_1 \sim_A \alpha_2$ implies $\alpha_1 = \alpha_2$. (The classification in Example 4.2 is both separated and extensional whereas **Webster** is extensional.)

Example 4.4. Given any set A, the *powerset classification* associated with A is the classification whose types are subsets of A, whose tokens are elements of A and whose classification relation is defined by $a \models \alpha$ if and only if $a \in \alpha$. The classification is both extensional and separated.

Example 4.5. Given any function $f : A \to B$, the *map classification* associated with f is the classification whose types are elements of B, whose tokens are elements of A and whose classification relation is defined by $a \models b$ if and only if $b = f(a)$.

Example 4.6. Given a first-order language L, the *truth classification* of L is the classification whose types are sentences of L, whose tokens are L-structures, and whose classification relation is defined by $M \models \varphi$ if and only if φ is true in the structure M.[1] The type set of a token M is the set of all sentences of L true in M, usually called the *theory of* M. The token set of a sentence φ is the collection of all models of φ.

Definition 4.7. Classifications A and B are said to be *isomorphic*, written $A \cong B$, if there are one-to-one correspondences between their types and between their tokens such that given corresponding tokens a and b and corresponding types α and β, $a \models_A \alpha$ if and only if $b \models_B \beta$.

[1] Strictly speaking, this is not a classification because every language has a proper class of structures. We shall not be concerned with the matter of size here; with the usual modifications, everything carries over to classifications with classes of tokens and types.

Exercises

4.1. For Examples 4.4–4.6 what is the type set of each token and the token set of each type? Which of these classifications are in general separated or extensional? For those that are not one or the other, state the conditions under which they would be.

4.2 Infomorphisms

Now that the notion of a classification has been introduced, we turn to the main notion used in this book, that of an infomorphism. Infomorphisms are important relationships between classifications $\boldsymbol{A}$ and $\boldsymbol{B}$ and provide a way of moving information back and forth between them. The classifications can be of the same objects or they can be of different objects. When modeling a distributed system and one of its components, we typically think of the latter. But when we think of one object viewed from different perspectives, that of different people, different linguistic communities, different time zones, or different branches of science, it is the former we have in mind.

We shall work extensively with pairs $f = \langle f^{\wedge}, f^{\vee}\rangle$ of functions, of which $f^{\wedge}$ is a function from the types of one of these classifications to the types of the other, and $f^{\vee}$ is a function from the tokens of one of these classifications to the tokens of the other. To remember which function is which, recall that classifications are pictured with the types above the tokens. The function $f^{\wedge}$ (read "f up") on types has the caret pointing up, the function $f^{\vee}$ (read "f down") on tokens has the caret pointing down.

We are primarily interested in the case where the two functions map in opposite directions. We say that f is a *contravariant* pair from $\boldsymbol{A}$ to $\boldsymbol{B}$, and write $f : \boldsymbol{A} \rightleftarrows \boldsymbol{B}$, if $f^{\wedge}: \mathrm{typ}(\boldsymbol{A}) \to \mathrm{typ}(\boldsymbol{B})$ and $f^{\vee}: \mathrm{tok}(\boldsymbol{B}) \to \mathrm{tok}(\boldsymbol{A})$. In later lectures we shall explore the *covariant* case, $f : \boldsymbol{A} \rightrightarrows \boldsymbol{B}$, in which the functions both run the same direction: $f^{\wedge}: \mathrm{typ}(\boldsymbol{A}) \to \mathrm{typ}(\boldsymbol{B})$ and $f^{\vee}: \mathrm{tok}(\boldsymbol{A}) \to \mathrm{typ}(\boldsymbol{B})$.

Definition 4.8. *An infomorphism* $f : \boldsymbol{A} \rightleftarrows \boldsymbol{B}$ *from* $\boldsymbol{A}$ *to* $\boldsymbol{B}$ *is a contravariant pair of functions* $f = \langle f^{\wedge}, f^{\vee}\rangle$ *satisfying the following Fundamental Property of Infomorphisms:*

$$f^{\vee}(b) \models_{\boldsymbol{A}} \alpha \quad \text{iff} \quad b \models_{\boldsymbol{B}} f^{\wedge}(\alpha)$$

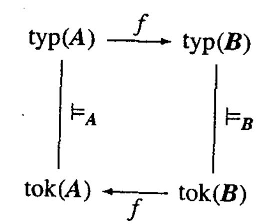
for each token $b \in \mathrm{tok}(\boldsymbol{B})$ *and each type* $\alpha \in \mathrm{typ}(\boldsymbol{A})$.[2]

[2] Strictly speaking, an infomorphism consists of two things: a pair of classifications $\langle \boldsymbol{A}, \boldsymbol{B}\rangle$ and a contravariant pair $f = \langle f^{\wedge}, f^{\vee}\rangle$ of functions between $\boldsymbol{A}$ and $\boldsymbol{B}$, satisfying the above condition.

We will omit the "up" and "down" superscripts if no confusion is likely, in which case the fundamental property reads

$$f(b) \models_{\boldsymbol{A}} \alpha \quad \text{iff} \quad b \models_{\boldsymbol{B}} f(\alpha).$$

We use classification diagrams to depict infomorphisms as follows, again leaving off the superscripts on f because it is clear which function is which:



Example 4.9. Here is an infomorphism that represents the way that the punctuation at the end of a written sentence carries information about the type of the sentence. We have two classifications, ***Punct*** and ***Sent***. The tokens of ***Punct*** are written inscriptions of the punctuation marks of English, things like commas, periods, and question marks. We will classify these marks in the way just suggested, by PERIOD, EXCLAMATION MARK, QUESTION MARK, COMMA, and so forth. The tokens of ***Sent*** are full, written inscriptions of grammatical sentences of English. For types we take three: DECLARATIVE, QUESTION, and OTHER.

Define an infomorphism f : ***Punct*** $\rightleftarrows$ ***Sent*** as follows. On tokens, f assigns to each sentence token its own terminating punctuation mark. On types, f assigns DECLARATIVE to EXCLAMATION MARK and PERIOD, QUESTION to QUESTION MARK, and OTHER to the other types of ***Punct***. The fundamental property of infomorphisms is the requirement that a written token be of the type indicated by its own punctuation. This condition is satisfied if we treat commands as declarative sentences and if part of what we mean for a written token of an English sentence to be grammatical is that it end with the right punctuation mark.

Example 4.10. Let $\boldsymbol{A}$ and $\boldsymbol{B}$ be power classifications on two sets A and B, respectively. Let us examine the meaning of the infomorphism condition in this case. By definition, an infomorphism $f : \boldsymbol{A} \rightleftarrows \boldsymbol{B}$ consists of a function $f^{\vee}: B \to A$ and a function $f^{\wedge}: \mathrm{pow}\, A \to \mathrm{pow}\, B$ satisfying the condition that for all $b \in B$ and all $\alpha \subseteq A$,

$$f^{\vee}(b) \models_{\boldsymbol{A}} \alpha \quad \text{iff} \quad b \models_{\boldsymbol{B}} f^{\wedge}(\alpha),$$

which is to say that

$$f^{\vee}(b) \in \alpha \quad \text{iff} \quad b \in f^{\wedge}(\alpha).$$

But this line says that

$$f^{\wedge}(\alpha) = \{b \in B \mid f^{\vee}(b) \in \alpha\},$$

which is, by definition, $f^{\vee -1}[\alpha]$. In other words, in the case of power classifications, an infomorphism is nothing but a function from B into A paired with its inverse, the latter being a function from subsets of A to subsets of B. This is, of course, a familiar situation in mathematics. For example, in topology we use open sets to classify points and focus on "continuous" functions, functions whose inverse takes open sets to open sets.

We said earlier that we think of an infomorphism $f : \mathbf{A} \rightleftarrows \mathbf{B}$ as going from $\mathbf{A}$ to $\mathbf{B}$. Example 4.10 suggests that this is backward, that it goes from $\mathbf{B}$ to $\mathbf{A}$. The direction of the infomorphism in 4.8 is a terminological decision that could have gone either way. Our choice was motivated by the following example from logic.

Example 4.11 (Interpretations in First-Order Logic). Let L and L' be first-order languages. We assume for simplicity that the basic symbols of L are relation symbols $R_1, \ldots, R_n$ of arities $\nu(R_1), \ldots, \nu(R_n)$, respectively.[3] An *interpretation* of L into L' is determined by a formula $\psi(x)$ of L' with the one free variable x (used to define the range of quantification) and formulas $\varphi_1, \ldots, \varphi_n$ of L' such that φ_i has free variables $x_1, \ldots, x_{\nu(R_i)}$, where φ_i is used to interpret R_i.

Such an interpretation I can be thought of as an infomorphism on the truth classifications of L and L' in a natural way. On types, that is on L-sentences, we get a mapping $f^{\wedge}$ from L-sentences to L'-sentences as follows. Each atomic formula $R_i(t_1, \ldots, t_{\nu(R_i)})$ is replaced by $\varphi(t_1, \ldots, t_{\nu(R_i)})$ and all quantifiers are relativized to ψ.[4] For example, the formula

$$\exists y R_2(x, y)$$

is mapped to the formula

$$\exists y(\psi(y) \wedge \varphi_2(x, y)),$$

[3] The arity of a relation is just its number of places. So, for example, if R is binary, then $\nu(R) = 2$.
[4] Some care has to be taken to avoid clashes of free and bound variables, but we will not worry about the details here.

and the sentence

$$\forall x[R_1(x) \rightarrow \exists y R_2(x, y)]$$

is mapped to the sentence

$$\forall x[\psi(x) \rightarrow [\varphi_1(x) \rightarrow \exists y(\psi(y) \wedge \varphi_2(x, y))]].$$

On structures, the interpretation works in the opposite direction. Given an L'-structure M', we use the formulas of L' to define a model $f^{\vee}(M')$. Its domain consists of those objects b of M' that satisfy ψ in M'; for $1 \le i \le n$, the interpretation of R_i in $f^{\vee}(M')$ is the set of $\nu(R_i)$-ary sequences that satisfy $\varphi_i(x_1, \ldots, x_{\nu(R_i)})$ in M'.

It is a straightforward matter to check that for each model M' of L' and each sentence φ of L, φ is true in $f^{\vee}(M')$ if and only if $f^{\wedge}(\varphi)$ is true in M'. (It would be a good exercise to verify this.) Thus the contravariant pair $\langle f^{\wedge}, f^{\vee}\rangle$ is an infomorphism from the truth classification of L to the truth classification of L'.

Definition 4.12. For any classification $\mathbf{A}$, the *identity* infomorphism $1_A : \mathbf{A} \rightleftarrows \mathbf{A}$ consists of the identity functions on types and tokens, respectively.

Justification. In this definition, we have implicitly made a claim, namely, that 1_A is indeed an infomorphism. In this case, the claim is obvious. In other definitions, we will be making similar implicit claims and so will need to justify them. □

Definition 4.13. Given infomorphisms $f : \mathbf{A} \rightleftarrows \mathbf{B}$ and $g : \mathbf{B} \rightleftarrows \mathbf{C}$, the *composition* $gf : \mathbf{A} \rightleftarrows \mathbf{C}$ of f and g is the infomorphism defined by $(gf)^{\wedge} = g^{\wedge} f^{\wedge}$ and $(gf)^{\vee} = f^{\vee} g^{\vee}$.

Justification. We leave it to the reader to verify that the composition of infomorphisms really is an infomorphism. □

An infomorphism $f : \mathbf{A} \rightleftarrows \mathbf{B}$ is an *isomorphism* if both $f^{\wedge}$ and $f^{\vee}$ are bijections. Classifications $\mathbf{A}$ and $\mathbf{B}$ are isomorphic if and only if there is an isomorphism between them.

4.3 Information Channels Defined

We now come to our key proposal for modeling information flow in distributed systems.

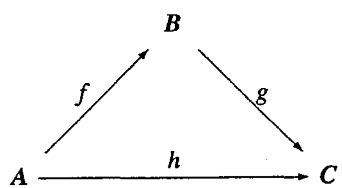
Definition 4.14. A *channel* $\mathcal{C}$ is an indexed family $\{f_i : \boldsymbol{A}_i \rightleftarrows \boldsymbol{C}\}_{i \in I}$ of infomorphisms with a common codomain $\boldsymbol{C}$, called the *core* of $\mathcal{C}$. The tokens of $\boldsymbol{C}$ are called *connections*; a connection c is said to *connect* the tokens $f_i(c)$ for $i \in I$. A channel with index set $\{0, \ldots, n-1\}$ is called an *n-ary* channel.

When we need a notation for the various parts of a channel $\mathcal{C}$, we write $\mathcal{C} = \{h_i^c : \boldsymbol{A}_i \rightleftarrows \text{core}(\mathcal{C})\}_{i \in I}$.

As we have explained in Part I, our proposal is that information flow in a distributed system is relative to some conception of that system as an information channel.

Refinements

We sometimes draw diagrams showing infomorphisms between several classifications. For the sake of graphical simplicity, we draw an arrow only in the direction of the function on types. Think of an infomorphism $f : \boldsymbol{A} \rightleftarrows \boldsymbol{B}$ as a way of translating from the types of $\boldsymbol{A}$ to those of $\boldsymbol{B}$. For example, the diagram

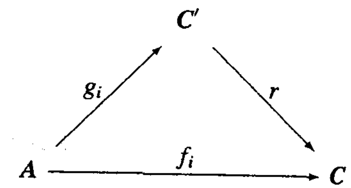


indicates that $f, g,$ and h are contravariant pairs as follows: $f : \boldsymbol{A} \rightleftarrows \boldsymbol{B}$, $g : \boldsymbol{B} \rightleftarrows \boldsymbol{C}$, and $h : \boldsymbol{A} \rightleftarrows \boldsymbol{C}$.

To say that this diagram commutes is to assert that $h = gf$. For types, this means that $h(\alpha) = g(f(\alpha))$, for all $\alpha \in \text{typ}(\boldsymbol{A})$, whereas for tokens it means that $h(c) = f(g(c))$ for all tokens $c \in \text{tok}(\boldsymbol{C})$. (Notice the different order of f and g in these two equations.)

We can now define what it means for one channel to refine another.

Definition 4.15. Let $\mathcal{C} = \{f_i : \boldsymbol{A}_i \rightleftarrows \boldsymbol{C}\}_{i \in I}$ and $\mathcal{C}' = \{g_i : \boldsymbol{A}_i \rightleftarrows \boldsymbol{C}'\}_{i \in I}$ be channels with the same component classifications $\boldsymbol{A}_i$. A *refinement* infomorphism r from $\mathcal{C}'$ to $\mathcal{C}$ is an infomorphism $r : \boldsymbol{C}' \rightleftarrows \boldsymbol{C}$ such that for each i, $f_i = rg_i$, that is, such that the following diagram commutes:

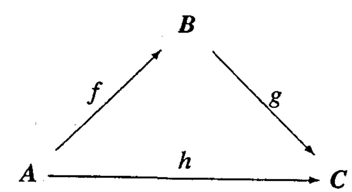


The channel $\mathcal{C}'$ is a *refinement* of the channel $\mathcal{C}$ if there is a refinement r from $\mathcal{C}'$ to $\mathcal{C}$.

Example 4.16. One example of a refinement is where $\text{typ}(\boldsymbol{C}) = \text{typ}(\boldsymbol{C}')$, $\text{tok}(\boldsymbol{C}) \subseteq \text{tok}(\boldsymbol{C}')$, and their classification relations agree on their common tokens. The refinement infomorphism is the identity function on types and the inclusion function on tokens. This refinement amounts to taking more tokens into account in the more refined channel. The more refined channel gives more reliable information, though at a cost, as we will see.

While on the subject of diagrams, we prove a lemma that is useful for determining that a pair of functions is an infomorphism. We say that a pair $f : \boldsymbol{A} \rightleftarrows \boldsymbol{B}$ of functions is *token surjective* (or *type surjective*) if $f^{\vee}$ (or $f^{\wedge}$, respectively) is surjective.

Lemma 4.17. *Let $f : \boldsymbol{A} \rightleftarrows \boldsymbol{B}$, $g : \boldsymbol{B} \rightleftarrows \boldsymbol{C}$, and $h : \boldsymbol{A} \rightleftarrows \boldsymbol{C}$ be contravariant pairs such that the following diagram commutes:*



1. *If f and g are infomorphisms, so is h.*
2. *If f and h are infomorphisms, and f is type surjective, then g is an infomorphism.*
3. *If g and h are infomorphisms, and g is token surjective, then f is an infomorphism.*

Proof. (1) is straightforward. To prove (2) let $\beta \in \text{typ}(\boldsymbol{B})$ and $c \in \text{tok}(\boldsymbol{C})$. We need to show that $g(c) \models_{\boldsymbol{B}} \beta$ if and only if $c \models_{\boldsymbol{C}} g(\beta)$. Because $f^{\wedge}$ is surjective, we know that $\beta = f(\alpha)$ for some $\alpha \in \text{typ}(A)$. The following are then equivalent:

$$\begin{aligned} g(c) \models_B \beta \quad &\text{iff} \quad g(c) \models_B f(\alpha) \text{ (because } \beta = f(\alpha)) \\ &\text{iff} \quad f(g(c)) \models_A \alpha \text{ (because } f \text{ is an infomorphism)} \\ &\text{iff} \quad h(c) \models_A \alpha \quad \text{(because } h = gf) \\ &\text{iff} \quad c \models_C h(\alpha) \quad \text{(because } h \text{ is an infomorphism)} \\ &\text{iff} \quad c \models_C g(f(\alpha)) \text{ (because } h = gf) \\ &\text{iff} \quad c \models_C g(\beta) \quad \text{(because } \beta = f(\alpha)) \end{aligned}$$

(3) is proved in a similar way. (It also follows from (2) by the duality of types and tokens discussed in the next section.) □

Exercises

4.2. If $f: A \rightleftarrows B$ is an infomorphism and b is indistinguishable from b' in B, show that $f(b)$ is indistinguishable from $f(b')$ in A.

4.3. Let A and B be classifications, and let $g: \text{tok}(B) \to \text{tok}(A)$. Find a necessary and sufficient condition for there to be an infomorphism $f: A \rightleftarrows B$ with $f^\vee = g$.

4.4. There is a special classification $\mathbf{o}$ that plays the role of a zero classification. It has a single token but no types. Show that for any classification A there is a unique infomorphism $f: \mathbf{o} \rightleftarrows A$. (Note that any two classifications with this property are isomorphic, so $\mathbf{o}$ is unique up to isomorphism.)

4.5. Given a separated classification A and infomorphisms $f: A \rightleftarrows B$ and $g: A \rightleftarrows B$, show that if $f^\wedge = g^\wedge$, then $f = g$.

4.6. (†) The category whose objects are classifications and whose morphisms are infomorphisms is sometimes called the *category of Chu spaces*, or the "Chu over Set" category. Show that an infomorphism $f: A \rightleftarrows B$ is

1. monic in this category if and only if $f^\wedge$ is injective and $f^\vee$ is surjective and
2. epi in this category if $f^\wedge$ is surjective and $f^\vee$ is injective.

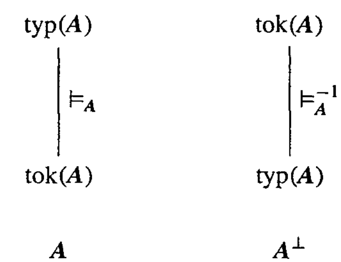
4.4 Type-Token Duality

In any classification, we think of the types as classifying the tokens, but it is often useful to think of the tokens as classifying their types. Elements of a set can classify those subsets containing them; models can classify the sentences they make true; objects can be regarded as classifying their properties.

Definition 4.18.

1. For any classification A, the *flip* of A, is the classification $A^\perp$ whose tokens are the types of A, whose types are the tokens of A and such that $\alpha \models_{A^\perp} a$ if and only if $a \models_A \alpha$.
2. Given any pair of functions $f: A \rightleftarrows B$, define $f^\perp: B^\perp \rightleftarrows A^\perp$ by $f^{\perp\wedge} = f^\vee$ and $f^{\perp\vee} = f^\wedge$.

Pictorially, $A^\perp$ amounts to just flipping A upside down (hence the name "flip" of A, as follows (where $\models_A^{-1}$ represents the converse of $\models_A$):



In terms of classification tables, flipping amounts to interchanging rows and columns.

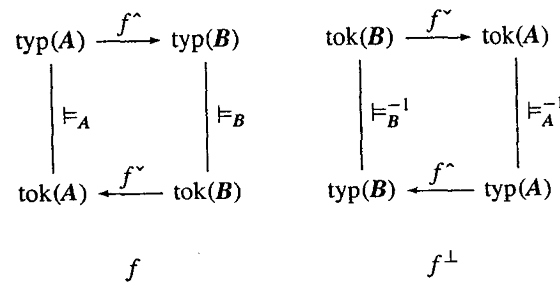
When deciding which way to model a classification we sometimes turn to epistemological considerations. The types are usually considered as "given"; they are things we know about. The tokens are things we want to find out about, i.e., about which we want information. For example, with objects and properties one usually wants to find out about the objects by classifying them in terms of familiar properties. In this case, the objects are taken as tokens and the properties as types. In other circumstances, one might want to learn about some properties by seeing which objects have them. For example, we use paradigmatic objects to explain a concept to a child unfamiliar with it. In this case, we would take the properties or concepts as tokens and the objects to be types. (Chapter 17 contains an application of this idea to the theory of speech acts.)

The observation that the flip of a classification is itself a classification, together with the fact that it behaves nicely with infomorphisms, sets up a duality between types and tokens that can cut certain work we have to do in half, because it means that any definition or notion involving types or tokens has a dual about tokens or types. For example, the dual of "the type set of a token" is "the token set of a type." The notions of coextensional types and indistinguishable tokens are duals of one another. Similarly, the dual of the notion of an extensional classification is that of a separated classification.

When we say that flipping is well-behaved with respect to infomorphisms, what we have in mind is the following:

Proposition 4.19. *$f : A \rightleftarrows B$ is an infomorphism if and only if $f^{\perp} : B^{\perp} \rightleftarrows A^{\perp}$ is an infomorphism.*

Proof. Looking at the following diagrams, we see one is just a redrawing of the other.



□

Proposition 4.20.

1. $(A^{\perp})^{\perp} = A$ *and* $(f^{\perp})^{\perp} = f$.
2. $(fg)^{\perp} = g^{\perp} f^{\perp}$.

Proof. This is a routine verification. □

As a simple example of duality at work, we can dualize Exercise 4.2 to obtain the following:

Proposition 4.21. *Given an infomorphism $f : A \rightleftarrows B$, if α is coextensive with α' in A, then $f(\alpha)$ is coextensive with $f(\alpha')$ in B.*

Proof. Apply Exercise 4.2 to $A^{\perp}$ and $f^{\perp}$. □

We will make frequent use of such dualities in what follows.

Exercise

4.7. Dualize Exercise 4.4.

Lecture 5

Operations on Classifications

There are many operations that take classifications and produce new classifications. This lecture discusses two of these operations that are quite important from an information-theoretic perspective. Whenever such an operation is introduced, we will study how it interacts with infomorphisms.

5.1 Sums of Classifications

As should be evident from Part I, the most basic way of combining classifications for our purposes is to add them. In forming a sum of classifications A and B, we classify pairs $\langle a, b\rangle$ of tokens, a from A and b from B, using types that are copies of the types in A and B.

Definition 5.1. Given classifications A and B, the *sum* $A + B$ of A and B is the classification defined as follows:

1. The set $\text{tok}(A + B)$ is the Cartesian product of $\text{tok}(A)$ and $\text{tok}(B)$. Specifically, the tokens of $A + B$ are pairs $\langle a, b\rangle$ of tokens, $a \in \text{tok}(A)$ and $b \in \text{tok}(B)$.
2. The set $\text{typ}(A + B)$ is the disjoint union of $\text{typ}(A)$ and $\text{typ}(B)$. For concreteness, the types of $A + B$ are pairs $\langle i, \alpha\rangle$, where $i = 0$ and $\alpha \in \text{typ}(A)$ or $i = 1$ and $\alpha \in \text{typ}(B)$.
3. The classification relation $\models_{A+B}$ of $A + B$ is defined by

$$\langle a, b\rangle \models_{A+B} \langle 0, \alpha\rangle \quad \text{iff} \quad a \models_A \alpha$$
$$\langle a, b\rangle \models_{A+B} \langle 1, \beta\rangle \quad \text{iff} \quad b \models_B \beta.$$

Example 5.2. Suppose that Sw is a classification of light switches using types UP and DN. Then $Sw + Sw$ classifies pairs $\langle s_1, s_2\rangle$ of switches using four types,

types we write informally as $\text{UP}_1, \text{UP}_2, \text{DN}_1$, and DN_2. For example, $\langle s_1, s_2\rangle$ is of type DN_2 if and only if s_2 is down.

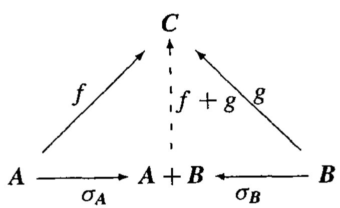
Definition 5.3. There are natural infomorphisms $\sigma_A : \boldsymbol{A} \rightleftarrows \boldsymbol{A} + \boldsymbol{B}$ and $\sigma_B : \boldsymbol{B} \rightleftarrows \boldsymbol{A} + \boldsymbol{B}$ defined as follows:

1. $\sigma_A(\alpha) = \langle 0, \alpha\rangle$ for each $\alpha \in \text{typ}(\boldsymbol{A})$,
2. $\sigma_B(\beta) = \langle 1, \beta\rangle$ for each $\beta \in \text{typ}(\boldsymbol{B})$, and
3. for each pair $\langle a, b\rangle \in \text{tok}(\boldsymbol{A} + \boldsymbol{B})$, $\sigma_A(\langle a, b\rangle) = a$ and $\sigma_B(\langle a, b\rangle) = b$.

Justification. It is routine to verify that σ_A and σ_B are infomorphisms. □

The following proposition will be used implicitly over and over again in what follows.

Proposition 5.4 (Universal Mapping Property for Sums). *Given infomorphisms $f : \boldsymbol{A} \rightleftarrows \boldsymbol{C}$ and $g : \boldsymbol{B} \rightleftarrows \boldsymbol{C}$, there is a unique infomorphism $f + g$ such that the following diagram commutes:*



Proof. Let us prove this explicitly, even though it is pretty clear. First, suppose we had such an infomorphism $f + g$. From the fact that the diagram commutes we see that $f + g$ must satisfy the following equations:

$$(f+g)\hat{}(\langle 0, \alpha\rangle) = f\hat{}(\alpha)$$
$$(f+g)\hat{}(\langle 1, \beta\rangle) = g\hat{}(\beta)$$
$$(f+g)\check{}(c) = \langle f\check{}(c), g\check{}(c)\rangle$$

But we can use these equations to define $f + g$. If we show that $f + g$ so defined is an infomorphism, we will show both existence and uniqueness. Suppose $c \in \text{tok}(\boldsymbol{C})$ and that $\langle i, \sigma\rangle \in \text{typ}(\boldsymbol{A} + \boldsymbol{B})$. We need to check that $(f + g)(c)$ is of type $\langle i, \sigma\rangle$ in $\boldsymbol{A} + \boldsymbol{B}$ if and only if c is of type $(f + g)(\langle i, \sigma\rangle)$ in $\boldsymbol{C}$. There are two cases to consider, depending on whether $i = 0$ or $i = 1$. They are symmetrical, though, so assume $i = 0$. Then the following are equivalent:

$(f + g)(c)$ is of type $\langle 0, \sigma\rangle$ in $\boldsymbol{A} + \boldsymbol{B}$;
$\langle f(c), g(c)\rangle$ is of type $\langle 0, \sigma\rangle$ in $\boldsymbol{A} + \boldsymbol{B}$ (by the definition of $(f + g)(c)$);

$f(c)$ is of type σ in $\boldsymbol{A}$ (by the definition of $\models_{A+B}$);
c is of type $f(\sigma)$ in $\boldsymbol{C}$ (by the fact that f is an infomorphism); and
c is of type $(f + g)(\langle 0, \sigma\rangle)$ in $\boldsymbol{C}$ (because $f(\sigma) = (f + g)(\langle 0, \sigma\rangle)$).

So we have the desired equivalence. □

Many of our constructions extend from two classifications to indexed families of classifications in a straightforward way. We give the details for addition and then are less detailed about others. By an *indexed family* $\{\boldsymbol{A}_i\}_{i\in I}$ of classifications we mean a function on some set I taking classifications as values. We call I the index set of the indexed family.

Definition 5.5. Let $\{\boldsymbol{A}_i\}_{i\in I}$ be an indexed family of classifications. The sum $\sum_{i\in I} \boldsymbol{A}_i$ of the family $\{\boldsymbol{A}_i\}_{i\in I}$ is defined by the following:

1. $\text{tok}(\sum_{i\in I} \boldsymbol{A}_i)$ is the Cartesian product of the sets $\text{tok}(\boldsymbol{A}_i)$ for $i \in I$;
2. $\text{typ}(\sum_{i\in I} \boldsymbol{A}_i)$ is the disjoint union of the sets $\text{typ}(\boldsymbol{A}_i)$ for $i \in I$; and
3. for each $\vec{a} \in \text{tok}(\sum_{i\in I} \boldsymbol{A}_i)$ and each $\alpha \in \text{typ}(\boldsymbol{A}_i)$, writing a_i for the ith component of $\vec{a}$,

$$\vec{a} \models \langle i, \alpha\rangle \quad \text{in} \sum_{i\in I} \boldsymbol{A}_i \quad \text{iff} \quad a_i \models_{A_i} \alpha.$$

This is a generalization of our previous definition because we can think of a pair $\boldsymbol{A}_1, \boldsymbol{A}_2$ of classifications as an indexed family $\{\boldsymbol{A}\}_{i\in\{1,2\}}$ for which the index set is $\{1, 2\}$. In this case, $\sum_{i\in\{1,2\}} \boldsymbol{A}_i = \boldsymbol{A}_1 + \boldsymbol{A}_2$.

Definition 5.6. Natural infomorphisms $\sigma_i : \boldsymbol{A}_i \rightleftarrows \sum_{i\in I} \boldsymbol{A}_i$ are defined as follows.

1. For each $\alpha \in \text{typ}(\boldsymbol{A}_i)$, $\sigma_i(\alpha)$ is the copy of α in $\text{typ}(\sum_{i\in I} \boldsymbol{A}_i)$.
2. For each $\vec{a} \in \text{tok}(\sum_{i\in I} \boldsymbol{A}_i)$, $\sigma_i(\vec{a}) = a_i$, where a_i is the ith component of a.

Finally, given a family $\{f_i : \boldsymbol{A}_i \rightleftarrows \boldsymbol{C}\}_{i\in I}$ of infomorphisms, the *sum* $\sum_{i\in I} f_i$ of this family is the unique infomorphism $h : \sum_{i\in I} \boldsymbol{A}_i \rightleftarrows \boldsymbol{C}$ such that $h\sigma_{A_i} = f_i$ for each $i \in I$.

Justification. This is a straightforward generalization of the finite case. □

Sums and Channels. The importance of the addition operation is evident from our examples in Part I. It gives us a way of putting the components of an information channel $\mathcal{C} = \{f_i : \boldsymbol{A}_i \rightleftarrows \boldsymbol{C}\}_{i\in I}$ together into a single classification, $\boldsymbol{A} = \sum_{i\in I} \boldsymbol{A}_i$, and the infomorphisms together into a single infomorphism

$f = \sum_{i \in I} f_i$, with $f : A \rightleftarrows C$. This greatly simplifying the kinds of channels we need to consider as well as giving a single classification on which we will discover a local logic that captures the constraints about the system implicit in the channel.

5.2 Invariants and Quotients

My copy of today's edition of the local newspaper bears much in common with that of my next door neighbor. If mine has a picture of President Clinton on page 2, so does hers. If mine has three sections, so does hers. For some purposes, it is convenient to identify these tokens. On the other hand, they are different, both as tokens and in terms of types. Mine has orange juice spilled on it, hers does not. Hers has the crossword puzzle solved, mine does not.

These observations have clear information-theoretic implications. I can get certain kinds of information about my neighbor's paper by looking at mine, but not other kinds. How can we view this in terms of classifications and infomorphisms?

Definition 5.7. Given a classification A, an *invariant* is a pair $I = \langle \Sigma, R \rangle$ consisting of a set $\Sigma \subseteq \mathrm{typ}(A)$ of types of A and a binary relation R between tokens of A such that if aRb, then for each $\alpha \in \Sigma$, $a \models_A \alpha$ if and only if $b \models_A \alpha$.

Example 5.8. Let A be the classification of newspapers by their physical properties. Let $I = \langle \Sigma, R \rangle$, where aRb hold if and only if a and b are physical copies of the same edition and Σ is the set of those properties that are invariant under R. This is an invariant on A. We want to form a new classification A/I that identifies tokens related by R, using types from Σ.

When we apply the notion of an invariant $I = \langle \Sigma, R \rangle$ the relation R will usually be an equivalence relation on the tokens of A. But it is convenient not to require this. After all, given any relation R on a set A, R is contained in a smallest equivalence relation $\equiv_R$ on A, the equivalence relation generated by R. This is simply the reflexive, symmetric, transitive closure of R. In the following definition we refer to the equivalence classes under $\equiv_R$ as the R-equivalence classes and write $[a]_R$ for the R-equivalence classes of the token a.

Definition 5.9. Let $I = \langle \Sigma, R \rangle$ be an invariant on the classification A. The *quotient of A by I*, written A/I, is the classification with types Σ, whose tokens are the R-equivalence classes of tokens of A, and with $[a]_R \models_{A/I} \alpha$ if and only if $a \models_A \alpha$.

Justification. A simple (inductive) proof shows that if $\langle \Sigma, R \rangle$ is an invariant on A, so is $\langle \Sigma, \equiv_R \rangle$. Hence if $a \equiv_R b$ and $\alpha \in \Sigma$, then $a \models_A \alpha$ if and only if $b \models_A \alpha$. Thus our definition of the relation $\models_{A/I}$ is well-defined. □

Here are some examples of this construction. We begin with a very simple example, move on to a more motivating example, and then turn to some important special cases.

Example 5.10. Recall the classification $A_{4.2}$ of Example 4.2. If we restrict attention to the types α_3 and α_4, we can discover a repeated pattern. The third and fourth rows look just like the first and second, respectively, and the sixth looks like the fifth. We can isolate this pattern as a new classification as follows. Let R be the relation given by $a_1 R a_3$, $a_2 R a_4$, and $a_5 R a_6$. There are three R-equivalence classes, one that groups a_1 with a_3, one that groups a_2 with a_4, and one that groups a_5 and a_6. Let us call these equivalence classes b_1, b_2, and b_3, respectively. Let $\Sigma = \{\alpha_3, \alpha_4\}$ and let $I = \langle \Sigma, R \rangle$. Our pattern tells us that I is an invariant so we can form a quotient classification $A_{4.2}/I$. This classification has as tokens b_1, b_2, b_3, types α_3 and α_4, and its classification relation is given by the following table:

	α_3	α_4
b_1	0	1
b_2	1	0
b_3	0	0

Example 5.11. Given a classification A, a binary relation R on tok(A) is an *indistinguishability relation* if all tokens related by R are indistinguishable in A. In that case, $\langle \mathrm{typ}(A), R \rangle$ is an invariant. The quotient of A by $\langle \mathrm{typ}(A), R \rangle$ is sometimes abbreviated as A/R.

Example 5.12. For any classification A, the pair $\langle \mathrm{typ}(A), \sim_A \rangle$ is an invariant. The corresponding quotient identifies tokens that are indistinguishable from one another in A and so is separated. It is called the *separated quotient* of A, written Sep(A).

Example 5.13. For any classification A, let $=_A$ be the identity relation on tok(A). For any set $\Sigma \subseteq \mathrm{typ}(A)$, the pair $\langle \Sigma, =_A \rangle$ is an invariant. The quotient of A by $\langle \Sigma, =_A \rangle$ is called the *restriction* of A to Σ, written as $A \mid \Sigma$.

Example 5.14. Given infomorphisms $f : A \rightleftarrows B$ and $g : A \rightleftarrows B$, we can obtain an invariant on A as follows. Let Σ_{fg} be the set of types $\alpha \in \mathrm{typ}(A)$ such that

$f(\alpha) = g(\alpha)$, and let R_{fg} be the binary relation between tokens of A defined by $a_1 R_{fg} a_2$ if and only if there is some $b \in \text{tok}(\boldsymbol{B})$ such that $f(b) = a_1$ and $g(b) = a_2$. The pair $\langle \Sigma_{fg}, R_{fg} \rangle$ is easily seen to be an invariant on $\boldsymbol{A}$.

Definition 5.15. Given a classification $\boldsymbol{A}$ and an invariant $I = \langle \Sigma, R \rangle$ on $\boldsymbol{A}$, the *canonical quotient infomorphism* $\tau_I : \boldsymbol{A}/I \rightleftarrows \boldsymbol{A}$ is the inclusion function on types, and on tokens, maps each token of $\boldsymbol{A}$ to its R-equivalence class.

Justification. From the definition of $\boldsymbol{A}/I$, $[a]_R \models_{\boldsymbol{A}/I} \alpha$ if and only if $a \models_{\boldsymbol{A}} \alpha$, and so τ_I is an infomorphism. □

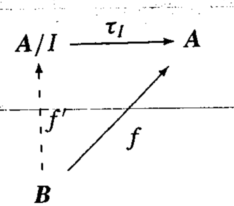
Definition 5.16. Given an invariant $I = \langle \Sigma, R \rangle$ on $\boldsymbol{A}$, an infomorphism $f : \boldsymbol{B} \rightleftarrows \boldsymbol{A}$ *respects* I if

1. for each $\beta \in \text{typ}(\boldsymbol{B})$, $f(\beta) \in \Sigma$; and
2. if $a_1 R a_2$, then $f(a_1) = f(a_2)$.

Example 5.17. Let $f : \boldsymbol{B} \rightleftarrows \boldsymbol{A}$ be any infomorphism. There is a natural invariant on $\boldsymbol{A}$ respected by f; namely, the types are the range of $f^\wedge$ and the relation is given by $a_1 R a_2$ if and only if $f(a_1) = f(a_2)$. (This invariant is called the *cokernel* of f.)

Given any invariant I, the canonical quotient infomorphism $\tau_I : \boldsymbol{A}/I \rightleftarrows \boldsymbol{A}$ clearly respects I. What is more, it does so canonically in the following sense.

Proposition 5.18. *Let I be an invariant on $\boldsymbol{A}$. Given any infomorphism $f : \boldsymbol{B} \rightleftarrows \boldsymbol{A}$ that respects I, there is a unique infomorphism $f' : \boldsymbol{B} \rightleftarrows \boldsymbol{A}/I$ such that the following diagram commutes.*



Proof. Let $I = \langle \Sigma, R \rangle$. Notice that if f' is an infomorphism making the above diagram commute then

1. $f'(\beta) = f(\beta)$ for each $\beta \in \text{typ}(\boldsymbol{B})$, and
2. $f'([a]_R) = f(a)$ for each $a \in \text{tok}(\boldsymbol{A}/I)$.

These conditions give us a definition of f'. That it is a good one follows from the assumption that f respects I: f' maps into the types of $\boldsymbol{A}/I$ because $f(\beta) \in A$ for each $\beta \in \text{typ}(\boldsymbol{B})$, and is well-defined on tokens because if $[a_1]_R = [a_2]_R$, then $R(a_1, a_2)$, so $f(a_1) = f(a_2)$. □

Definition 5.19. An infomorphism $f : \boldsymbol{A} \rightleftarrows \boldsymbol{B}$ is *token identical* if $\text{tok}(\boldsymbol{A}) = \text{tok}(\boldsymbol{B})$ and $f^\vee$ is the identity function on this set. Dually, f is *type identical* if $\text{typ}(\boldsymbol{A}) = \text{typ}(\boldsymbol{B})$ and $f^\wedge$ is the identity.

For example, given any indistinguishability relation R on $\boldsymbol{A}$, the quotient infomorphism $[]_R : \boldsymbol{A}/R \rightleftarrows \boldsymbol{A}$ is type identical.

Proposition 5.20. *Given classifications $\boldsymbol{A}$ and $\boldsymbol{B}$, if $\boldsymbol{A}$ is separated, then there is at most one type-identical infomorphism from $\boldsymbol{A}$ to $\boldsymbol{B}$.*

Proof. The result in an immediate corollary of Exercise 4.5. □

Dualizing Invariants and Quotients

There is a dual notion to that of an invariant and we will need this notion as well. Indeed, it will be quite important. We work through the process of dualizing the above, both because we need the notion and to give us a better feel for the uses of the type-token duality.

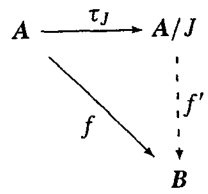
Given a classification $\boldsymbol{A}$, recall that an invariant is a pair $I = \langle \Sigma, R \rangle$ consisting of a set $\Sigma \subseteq \text{typ}(\boldsymbol{A})$ of types of $\boldsymbol{A}$ and a binary relation R between tokens of $\boldsymbol{A}$ such that if aRb, then for each $\alpha \in \Sigma$, $a \models_{\boldsymbol{A}} \alpha$ if and only if $b \models_{\boldsymbol{A}} \alpha$. By duality, a *dual invariant* is a pair $J = \langle A, R \rangle$ consisting of a set A of tokens and a binary relation R on types such that if $\alpha R \beta$, then for each $a \in A$, $a \models_{\boldsymbol{A}} \alpha$ if and only if $a \models_{\boldsymbol{A}} \beta$. The (dual) quotient of $\boldsymbol{A}$ by J has A for its set of tokens and has as types the R-equivalence classes of types of $\text{typ}(\boldsymbol{A})$. We write this as $\boldsymbol{A}/J$.

Continuing this process of dualization, given a classification $\boldsymbol{A}$ and a dual invariant $J = \langle A, R \rangle$ on $\boldsymbol{A}$, the *canonical quotient infomorphism* $\tau_J : \boldsymbol{A} \rightleftarrows \boldsymbol{A}/J$ is the inclusion function on tokens, and on types, maps each type of $\boldsymbol{A}$ to its R-equivalence class. An infomorphism $f : \boldsymbol{A} \rightleftarrows \boldsymbol{B}$ *respects* J if

1. for each $b \in \text{tok}(\boldsymbol{B})$, $f(b) \in A$, and
2. if $\alpha_1 R \alpha_2$, then $f(\alpha_1) = f(\alpha_1)$.

Dualizing Proposition 5.18 gives us the following proposition.

Proposition 5.21. *Let J be a dual invariant on A. Given any infomorphism $f : A \rightleftarrows B$ that respects J, there is a unique infomorphism $f' : A/J \rightleftarrows B$ such that the following diagram commutes:*



Exercises

5.1. Let A, B, and C be classifications. Prove that $A + \mathbf{o} \cong A$, $A + B \cong B + A$, and $A + (B + C) \cong (A + B) + C$ ($\mathbf{o}$ is the zero classification introduced in Exercise 4.4).

5.2. Show that a classification A is separated if and only if $A \cong \text{Sep}(A)$.

5.3. Define $I_1 \sqsubseteq I_2$ on invariants $I_1 = \langle \Sigma_1, R_1 \rangle$ and $I_2 = \langle \Sigma_2, R_2 \rangle$ if and only if $\Sigma_1 \subseteq \Sigma_2$ and $a R_2 b$ entails $a \equiv_{R_1} b$. Show that this is a preordering and that it is a partial ordering on invariants where the relation on tokens is an equivalence relation. Show that if $I_1 \sqsubseteq I_2$, then there is a natural infomorphism $f : A/I_1 \rightleftarrows A/I_2$.

5.4. Let $f : B \rightleftarrows A$ be an infomorphism. Show that the cokernel of f is the smallest invariant (in the sense of the $\sqsubseteq$-ordering) I such that f respects I.

5.5. (†) Show that the pair $I_{fg} = \langle \Sigma_{fg}, R_{fg} \rangle$ defined in Example 5.14 really is an invariant, and that an infomorphism $h : C \rightleftarrows A$ respects it if and only if $fh = gh$. Conclude that the canonical quotient infomorphism $\tau I_{fg} : A/I_{fg} \rightleftarrows A$ is the equalizer of f and g. Show that every canonical quotient infomorphism is an equalizer.

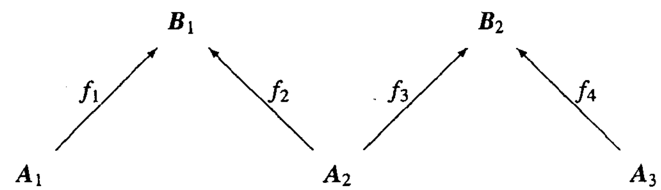
Lecture 6

Distributed Systems

We introduced the notion of an information channel $\mathcal{C} = \{f_i : A_i \rightleftarrows C\}_{i \in I}$ as a mathematical model of the intuitive idea of a distributed system, a whole made up of parts in a way that supports information flow. The core C of the channel represents the whole, the classifications A_i the parts, and the infomorphisms f_i the relationships between the whole and its parts. When we look at a typical real-world example, however, like that involving Judith from Lecture 1, we see that there are many interacting systems involved in information flow. The question arises as to whether we can put these various systems together and view it as a single channel. The aim of this lecture is to show that we can.

6.1 The Xerox Principle

To initiate the discussion, let us start with a special case, the one needed to justify Dretske's Xerox principle, the principle that says that information flow is transitive. To be more concrete, suppose we have two channels sharing a common component, maps. The first channel will involve people examining maps; this channel is to capture the process whereby a person's perceptual state carries information about the map at which the person is looking. The second channel will involve maps of mountains and other regions, the idea being that a map carries information about the region of which it is a map. Diagrammatically, we have the following picture:

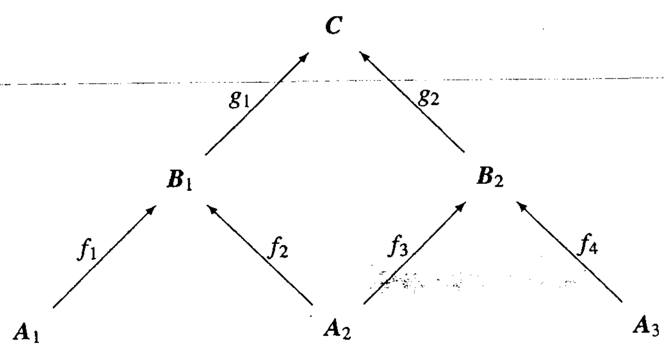


The connections of the first channel (the tokens of $\boldsymbol{B}_1$) are spatiotemporal perceptual events involving people (tokens in $\boldsymbol{A}_1$) looking at maps (tokens in $\boldsymbol{A}_2$). The connections of the second channel (the tokens of $\boldsymbol{B}_2$) are spatio-temporal events that involve the making of maps (tokens in $\boldsymbol{A}_2$) of various regions (tokens in $\boldsymbol{A}_3$).[1] We want to show that under certain circumstances, a person's perceptual state carries information about a particular mountain in virtue of the fact that the person is examining a map of that mountain. To do this, we need to see how to get a channel that puts these together into another channel.

There is a natural set of "connections" connecting people with regions: namely, go first to a person, then to a map she is examining, and then on to the map's region. Mathematically, we can model this with the set of pairs $\langle b_1, b_2\rangle$ such that $f_2(b_1) = f_3(b_2)$, say; that is, b_1 is to be a perceptual event, b_2 a map-making event, and they involve the same map $a_2 = f_2(b_1) = f_3(b_2)$.

How should we classify such connections $c = \langle b_1, b_2\rangle$? Because they are combinations of tokens from the classifications $\boldsymbol{B}_1$ and $\boldsymbol{B}_2$, we can certainly use the types of these two classifications to classify them. However, this alone misses something very important about their relationship. For suppose we have some type $\alpha_2 \in \text{typ}(\boldsymbol{A}_2)$ about maps. This type translates into some property $\beta_1 = f_2(\alpha_2)$ of perceptual events, but also translates into some other property $\beta_2 = f_3(\alpha_2)$ of mapmakings. If $f_2(b_1) = f_3(b_2) = a_2$, we know that the perceptual event b_1 is of type β_1 if and only if the map a_2 is of type α_2, but this latter holds if and only if the mapmaking event b_2 satisfies β_2. In other words, by restricting attention to connections $c = \langle b_1, b_2\rangle$ with $f_2(b_1) = f_3(b_2)$, the types β_1 and β_2 are equivalent because they are both "translations" of α_2. We need to build this into our information channel by identifying β_1 and β_2.

Thus we form a new classification $\boldsymbol{C}$ with the above tokens, and whose types are the types of $\boldsymbol{B}_1$ and $\boldsymbol{B}_2$, except that we identify types that come from a common type in $\boldsymbol{A}_2$. We can turn this into a channel connecting $\boldsymbol{B}_1$ to $\boldsymbol{B}_2$ by defining the infomorphisms in the obvious way, giving us the following picture:



[1] We will have more to say about these channels in the lecture on representations, Lecture 20.

Out of this picture we obtain a derived binary channel with core $\boldsymbol{C}$ linking $\boldsymbol{A}_1$ to $\boldsymbol{A}_3$. The infomorphisms of the channel are $h_1 = g_1 f_1$ and $h_3 = g_2 h_4$. This is the *sequential composition* of the original channels. A person a_1 is connected to a region a_3 relative to this channel if a_1 is examining some map of a_3. Of course there is also a ternary channel with the addition classification $\boldsymbol{A}_2$ and infomorphism $h_2 = g_1 f_2$. (Note that $h_2 = g_2 f_3$ because the diagram commutes.)

Notice that although we have used talk of people, maps, and regions to illustrate this construction, it is all perfectly general and applies to allow us to sequentially compose any pair of channels that line up as depicted in the first diagram above.

6.2 Distributed Systems to Channels

Having seen how to sequentially compose binary channels, we want to generalize this substantially.

Definition 6.1. A *distributed system* $\mathcal{A}$ consists of an indexed family $\text{cla}(\mathcal{A}) = \{\boldsymbol{A}_i\}_{i\in I}$ of classifications together with a set $\text{inf}(\mathcal{A})$ of infomorphisms all having both domain and codomain in $\text{cla}(\mathcal{A})$.

It is possible for $\boldsymbol{A}_i$ to be the same as $\boldsymbol{A}_j$ even if $i \neq j$. In other words, it is possible for one classification to be part of a distributed system in more than one way. This will be important in some of our applications. Also, there is no assumption that any infomorphisms in $\text{inf}(\mathcal{A})$ commute. This notion allows us to model distributed systems that are quite disparate, as in the example with Judith in the first lecture. We will show how any such system can be turned into a channel.[2]

Example 6.2. If we have a pair of binary channels as depicted above, it gives us a distributed system with five classifications and four infomorphisms.

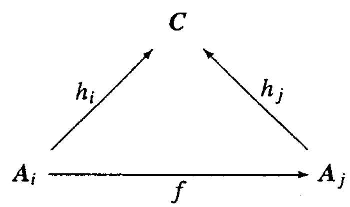
Example 6.3. Let Z be the set of time zones. We define a classification $\boldsymbol{A}_z$, for each $z \in Z$, as follows. The tokens of $\boldsymbol{A}_z$ are times (not their descriptions), the types of $\boldsymbol{A}_z$ are temporal descriptions, things like "12:05 A.M." The classification relation is given by $t \models_{\boldsymbol{A}_z} \alpha$ if and only if α is the correct description of t in time zone z. For example, the time right now is of type 12:05 A.M. in $\boldsymbol{A}_z$

[2] The reader versed in category theory will note that we could have required distributed systems to be subcategories of the category of all classifications with infomorphisms; this would not effect anything we do here, because it amounts to throwing in the identity infomorphisms and closing under composition.

if and only if it is now 12:05 A.M. in time zone z. There are lots of infomorphisms between these various classifications. For example, between $\boldsymbol{A}_e$ and $\boldsymbol{A}_p$ ("e" and "p" standing for Eastern and Pacific time zones in the United States, respectively), we have a token identical infomorphism $f_{ep} : \boldsymbol{A}_e \rightleftarrows \boldsymbol{A}_p$ given by subtracting three hours, mod 12, from the description, and changing AM to PM or vice versa when mod 12 is invoked; for example, f_{ep} of "12:05 A.M." is "9:05 P.M."

The relationship between distributed systems in general and channels is given by the following definition.

Definition 6.4. A channel $\mathcal{C} = \{h_i : \boldsymbol{A}_i \rightleftarrows \boldsymbol{C}\}_{i\in I}$ *covers* a distributed system $\mathcal{A}$ if $\text{cla}(\mathcal{A}) = \{\boldsymbol{A}_i\}_{i\in I}$ and for each $i, j \in I$ and each infomorphism $f : \boldsymbol{A}_i \rightleftarrows \boldsymbol{A}_j$ in $\inf(\mathcal{A})$, the following diagram commutes:



$\mathcal{C}$ is a *minimal cover* of $\mathcal{A}$ if it covers $\mathcal{A}$ and for every other channel $\mathcal{D}$ covering $\mathcal{A}$ there is a unique infomorphism from $\boldsymbol{C}$ to $\boldsymbol{D}$.

A minimal cover of a system $\mathcal{A}$ turns the whole distributed system into a channel. Minimal covers of arbitrary distributed systems may be constructed in a way that generalizes our first example. Minimal covers are known as "colimits" in category theory. The fact that colimits of classifications exist is a result owing to Barr and Chu; see Barr (1991) for a discussion of the history. We state this in the following way:

Theorem 6.5. *Every distributed system has a minimal cover, and it is unique up to isomorphism.*

We are interested not just in the existence of the minimal cover but in understanding its structure, so we take some pains over proving this theorem. We will construct one particular minimal cover of $\mathcal{A}$ and dub it the "limit" of $\mathcal{A}$, written $\lim \mathcal{A}$.[3] We begin with a few simple observations.

[3] Strictly speaking, we should call it the colimit of $\mathcal{A}$, of course.

Proposition 6.6. *Any two minimal covers of the same distributed system are isomorphic.*

Proof. For readers familiar with universal algebra or category theory, this proof will be routine. For others, we give an outline. The key to the proof is to realize that if $\mathcal{C}$ is a minimal cover, then the identity is the only refinement infomorphism from $\mathcal{C}$ to $\mathcal{C}$, because it is one, and there can be only one, by the definition of minimal cover. Now given this, suppose we had another minimal cover $\mathcal{D}$. Then there would be refinements r from $\mathcal{C}$ to $\mathcal{D}$ and r' from $\mathcal{D}$ to $\mathcal{C}$. But then the composition $r'r$ is a refinement infomorphism from $\mathcal{C}$ to $\mathcal{C}$ and so is the identity. Hence r and r' must be inverses of one another and so r is an isomorphism. □

Example 6.7. The sum of classifications is an example of a minimal cover. Given a family $\{\boldsymbol{A}_i\}_{i\in I}$ of classifications, let $\mathcal{A}$ be the distributed system consisting of these classifications and no infomorphisms. The channel with core $\sum_{i\in I} \boldsymbol{A}_i$ and infomorphisms $\sigma_{A_j} : \boldsymbol{A}_j \rightleftarrows \sum_{i\in I} \boldsymbol{A}_i$ for each $j \in I$ is a minimal cover of $\mathcal{A}$.

Example 6.8. The sequential composition of two binary channels constructed in the preceding section is a minimal cover for the distributed system consisting of five classifications and four infomorphisms.

We now generalize the construction from the special case of binary composition to arbitrary distributed systems. The construction will not yield exactly what we had before, but will yield something isomorphic to it.

Definition 6.9. Let $\mathcal{A}$ be a distributed system with classifications $\{\boldsymbol{A}_i\}_{i\in I}$. The *limit* $\lim \mathcal{A}$ *of* $\mathcal{A}$ is the channel constructed as follows.

1. The core of $\lim \mathcal{A}$ is a (dual) quotient of the *sum* $\boldsymbol{A} = \sum_{i\in I} \boldsymbol{A}_i$ of the classifications of the system. Write c_i for the ith coordinate of a token c of $\boldsymbol{A}$. Define the dual invariant $J = \langle C, R\rangle$ on $\boldsymbol{A}$ as follows. The set C of tokens consists of those tokens c of $\boldsymbol{A}$ such that $f(c_j) = c_i$ for each infomorphism $f : \boldsymbol{A}_i \rightleftarrows \boldsymbol{A}_j$ in $\inf(\mathcal{A})$. Intuitively, in terms of systems, this says we have a sequence of tokens that respects all possible whole-part relationships of the system. Define the relation R on the types of the sum by $\alpha R \alpha'$ if and only if there is an infomorphism $f : \boldsymbol{A}_i \rightleftarrows \boldsymbol{A}_j$ and a type $\alpha_0 \in \text{typ}(\boldsymbol{A}_i)$ such that $\alpha = \sigma_i(\alpha_0)$ and $\alpha' = \sigma_j(f(\alpha_0))$. Let $\boldsymbol{C}$ be the dual quotient $\sum_{i\in I} \boldsymbol{A}_i / J$.

2. To define the infomorphisms of $\lim \mathcal{A}$, let $g_j : \boldsymbol{A}_j \rightleftarrows \boldsymbol{C}$ (for any $j \in I$) be the pair of functions defined by

   $g_j(\alpha)$ is the R-equivalence class of $\sigma_j(\alpha)$ for each $\alpha \in \text{typ}(\boldsymbol{A}_j)$,

   and

   $g_j(c) = \sigma_j(c)$ for each $c \in \text{tok}(\boldsymbol{C})$.

   This is just a restriction of the usual infomorphism from a classification to its quotient and so g_j is an infomorphism.

Justification. To see that the definition is well defined, we must verify that J is indeed a dual invariant. Suppose that $c \in C$ and that $\alpha R \alpha'$. We need to verify that

$$c \models_{\boldsymbol{A}} \alpha \quad \text{iff} \quad c \models_{\boldsymbol{A}} \alpha'.$$

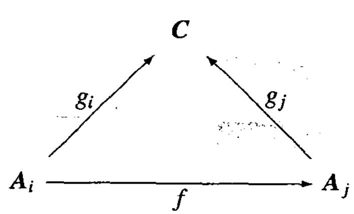
Because $\alpha R \alpha'$, there is an infomorphism $f : \boldsymbol{A}_i \rightleftarrows \boldsymbol{A}_j$ and a type $\alpha_0 \in \text{typ}(\boldsymbol{A}_i)$ such that $\alpha = \sigma_i(\alpha_0)$ and $\alpha' = \sigma_j(f(\alpha_0))$. Because $c \in C$, $f(c_j) = c_i$. Thus we have the following chain of equivalences:

$$\begin{aligned} c \models_{\boldsymbol{A}} \alpha \quad &\text{iff} \quad c \models_{\boldsymbol{A}} \sigma_i(\alpha_0) \\ &\text{iff} \quad c_i \models_{\boldsymbol{A}_i} \alpha_0 \\ &\text{iff} \quad f(c_j) \models_{\boldsymbol{A}_i} \alpha_0 \\ &\text{iff} \quad c_j \models_{\boldsymbol{A}_j} f(\alpha_0) \\ &\text{iff} \quad c \models_{\boldsymbol{A}} \sigma_j(f(\alpha_0)) \\ &\text{iff} \quad c \models_{\boldsymbol{A}} \alpha' \end{aligned}$$

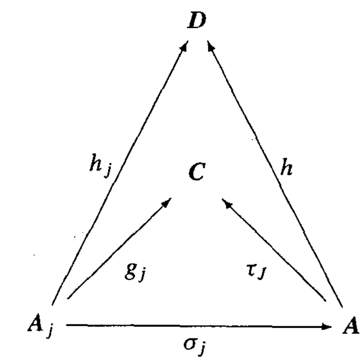
as desired. □

Theorem 6.10. *Given a distributed system $\mathcal{A}$, its limit* $\lim \mathcal{A}$ *is a minimal cover of $\mathcal{A}$.*

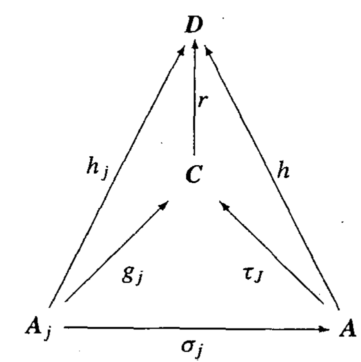
Proof. To see that $\lim \mathcal{A}$ is a cover, note that for each infomorphism $f : \boldsymbol{A}_i \rightleftarrows \boldsymbol{A}_j$ the following diagram commutes.



Now suppose $\mathcal{D} = \{h_i : \boldsymbol{A}_i \rightleftarrows \boldsymbol{D}\}_{i \in I}$ is a channel that also covers $\mathcal{A}$. Then for each $j \in I$, the following diagram commutes,



where $\boldsymbol{A} = \sum_{i \in I} \boldsymbol{A}_i$ and h is the sum of the infomorphisms h_j. Because h respects J, it factors uniquely through τ_J by the universal mapping property of the quotient $\boldsymbol{A}/J$. This gives us the following commuting diagram:



Thus r is the unique refinement infomorphism from Cha($\mathcal{A}$) to $\mathcal{D}$, as illustrated by the upper right triangle in the diagram. □

If one specializes the limit construction to the special case of sequential composition of binary channels considered in the discussion of the Xerox principle, the set of tokens is different. Instead of the pairs $\langle b_1, b_2 \rangle$ satisfying $f_2(b_1) = f_3(b_2)$ that we used there, we now find 5-tuples $\langle a_1, b_1, a_2, b_2, a_3 \rangle$ such that $a_1 = f_1(b_1)$, $f_2(b_1) = a_2 = f_3(b_2)$, and $a_3 = f_2(b_2)$. However, there is clearly a natural one-to-one correspondence between the two sets of tokens. Going one way, just throw out the a_i. Going the other way, insert the a_i as dictated by the f_i.

As a word of warning, we note that it is possible for the core of the limit of a distributed system to have an empty set of tokens. This happens when there are no sequences of tokens that respect all the infomorphisms of the system. Such a distributed system is called *completely incoherent.*

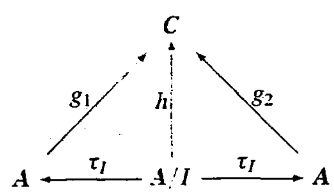
Quotient Channels

We initiated the discussion of invariants and quotients by an example involving the information flow between two copies of the same edition of a newspaper, noting how some information about one carries over intact to the other, but not all information carries. We never did really finish the discussion, though. We can do that now by taking the limit of a suitable distributed system.

Definition 6.11. Given an invariant $I = \langle \Sigma, R \rangle$ on a classification A, the *quotient channel of* A *by* I is the limit of the distributed system depicted by the following:

$$A \xleftarrow{\tau_I} A/I \xrightarrow{\tau_I} A$$

Notice that this distributed system has three classifications, not just two, even though two are the same. (This is made possible by the use of indexed families.) Similarly, there are two infomorphisms, each a copy of the canonical quotient infomorphism. The quotient channel makes the following diagram commute and is a refinement of any other such channel:



The quotient channel is easily seen to be isomorphic to the following. For each type $\alpha \in \text{typ}(A) - \Sigma$, there are two copies, say α_1 and α_2; let typ(C) consist of all these types, plus the types in Σ. The tokens of C consist of triples $c = \langle a_1, b, a_2 \rangle$, where $[a_1]_R = b = [a_2]_R$. The classification relation is defined for such $c = \langle a_1, b, a_2 \rangle$ by

$$\begin{array}{llll} c \models_C \alpha & \text{iff} & b \models_{A/I} \alpha & \text{for} \quad \alpha \in \Sigma \\ c \models_C \alpha_1 & \text{iff} & a_1 \models_A \alpha & \text{for} \quad \alpha \in \text{typ}(A) - \Sigma \\ c \models_C \alpha_2 & \text{iff} & a_2 \models_A \alpha & \text{for} \quad \alpha \in \text{typ}(A) - \Sigma \end{array}$$

The obvious infomorphisms $g_i : A_i \rightleftarrows C$ and $h : A/I \rightleftarrows C$ are defined by the following: each infomorphism is an inclusion on types, whereas on a token $c = \langle a_1, b, a_2 \rangle$, $f_i(c) = a_i$, and $h(c) = b$.

Now suppose that c connects a_1, b, and a_2. Then for any type $\alpha \in \Sigma$, we have $a_1 \models_A \alpha$ if and only if $b \models_{A/I} \alpha$ if and only if $a_2 \models_A \alpha$. On the other hand, if $\alpha \in \text{typ}(A) - \Sigma$, then $a_1 \models_A \alpha$ carries no information at all about b or a_2 via this channel.

Example 6.12. To finish up our newspaper example (5.8), let A and $I = \langle \Sigma, R \rangle$ be as in Example 5.8 and let C be its quotient channel. The equivalence class $[a]_R$ models the edition of a so a token $c = \langle a_1, b, a_2 \rangle$ of C is a model of the connection that exists between a_1 and a_2 if they are of the same edition. We see that for types in $\alpha \in \Sigma$, information about a particular newspaper a_1 carries information both about its edition, $[a_1]_R$, as well as about any other token a_2 of that same edition. That is, a_1 being of type α carries the information that a_2 is of the same type, relative to this channel, as long as $\alpha \in \Sigma$. Notice that this does not hold for $\alpha \in \text{typ}(A) - \Sigma$. Having an orange juice stain would be such a type.

Exercises

6.1. What are the infomorphisms g_1 and g_2 in the map example?

6.2. Determine the limit of the time-zone distributed system. Obtain a simpler, isomorphic minimal cover of this system. Interpret your results in terms of time zones.

6.3. Define a notion of *parallel composition* of binary channels.

6.4. Give an example of a completely incoherent distributed system.

Lecture 7

Boolean Operations and Classifications

The notion of a classification does not build in any assumptions about closure under the usual Boolean operations. It is natural to ask What role do the usual Boolean connectives play in information and its flow? This lecture takes an initial step toward answering this question. We will return to this question in later chapters as we develop more tools. It is not a central topic of the book, but it is one that needs to be addressed in a book devoted to the logic of information flow.

Actually, there are two ways of understanding Boolean operations on classifications. There are Boolean operations mapping classifications to classifications, and there are Boolean operations internal to (many) classifications. Because there is a way to explain the latter in terms of the former, we first discuss the Boolean operations on classifications.

7.1 Boolean Operations on Classifications

Given a set Φ of types in a classification, it is often useful to group together the class of tokens that are of *every* type in Φ. In general, there is no type in the classification with this extension. As a remedy, we can always construct a classification in which such a type exists. Likewise, we can construct a classification in which there is a type whose extension consists of all those tokens that are of at least one of the types in Φ.

Definition 7.1. Given a classification $\boldsymbol{A}$, the *disjunctive power* of $\boldsymbol{A}$, written $\vee\boldsymbol{A}$, and the *conjunctive power* of $\boldsymbol{A}$, written $\wedge\boldsymbol{A}$, are classifications specified as follows. Each of these classifications has the same tokens as $\boldsymbol{A}$ and has pow(typ($\boldsymbol{A}$)) as type set. Given $a \in \text{tok}(\boldsymbol{A})$ and $\Phi \subseteq \text{typ}(\boldsymbol{A})$,

1. $a \vDash_{\vee A} \Phi$ if and only if $a \vDash_A \sigma$ for some $\sigma \in \Phi$;
2. $a \vDash_{\wedge A} \Phi$ if and only if $a \vDash_A \sigma$ for *every* $\sigma \in \Phi$.

Proposition 7.2. *For each classification $\boldsymbol{A}$, there are natural embeddings $\eta^d_A : \boldsymbol{A} \rightleftarrows \vee\boldsymbol{A}$ and $\eta^c_A : \boldsymbol{A} \rightleftarrows \wedge\boldsymbol{A}$ defined by*

1. $\eta^d_A(\alpha) = \eta^c_A(\alpha) = \{\alpha\}$ *for each* $\alpha \in \text{typ}(\boldsymbol{A})$ *and*
2. $\eta^d_A(a) = \eta^c_A(a) = a$ *for each* $a \in \text{tok}(\vee\boldsymbol{A}) = \text{tok}(\wedge\boldsymbol{A})$.

Proof. Both η^d_A and η^c_A are clearly injective. They are infomorphisms because $a \vDash_A \alpha$ if and only if $a \vDash_{\vee A} \{\alpha\}$ if and only if $a \vDash_{\wedge A} \{\alpha\}$. □

Another useful operation on classifications is negation.

Definition 7.3. Given a classification $\boldsymbol{A}$, the *negation of* $\boldsymbol{A}$, $\neg\boldsymbol{A}$ is the classification with the same tokens and types as $\boldsymbol{A}$ such that for each token a and type α,

$$a \vDash_{\neg A} \alpha \quad \text{iff} \quad a \nvDash_A \alpha.$$

These operations on classifications have counterparts on infomorphisms.

Definition 7.4. Let $f : \boldsymbol{A} \rightleftarrows \boldsymbol{B}$ be an infomorphism. We define infomorphisms $\vee f : \vee\boldsymbol{A} \rightleftarrows \vee\boldsymbol{B}$, $\wedge f : \wedge\boldsymbol{A} \rightleftarrows \wedge\boldsymbol{B}$, and $\neg f : \neg\boldsymbol{A} \rightleftarrows \neg\boldsymbol{B}$ as follows: All three of these agree with f on tokens. On types they are defined by

1. $\vee f(\Theta) = f[\Theta]$ for all $\Theta \subseteq \text{typ}(\boldsymbol{A})$;
2. $\wedge f(\Theta) = f[\Theta]$ for all $\Theta \subseteq \text{typ}(\boldsymbol{A})$; and
3. $\neg f(\alpha) = f(\alpha)$

Justification. We need to verify that these are indeed infomorphisms. We check the first. We need to check that for all $b \in \text{tok}(\boldsymbol{B})$ and all $\Theta \subseteq \text{typ}(\boldsymbol{A})$, $f(b) \vDash_{\vee A} \Theta$ if and only if $b \vDash_{\vee B} \vee f(\Theta)$. This is seen by the following chain of equivalences:

$$\begin{array}{lll} f(b) \vDash_{\vee A} \Theta & \text{iff} & f(b) \vDash_A \alpha \text{ for some } \alpha \in \Theta \\ & \text{iff} & b \vDash_B f(\alpha) \text{ for some } \alpha \in \Theta \\ & \text{iff} & b \vDash_B \beta \text{ for some } \beta \in f[\Theta] \\ & \text{iff} & b \vDash_{\vee B} (\vee f)(\Theta) \end{array}$$

The others are similar and are left to the reader. □

Another Duality

Notice that for each classification $\boldsymbol{A}$, $\neg\neg\boldsymbol{A} = \boldsymbol{A}$. Consequently, the negation operation on classifications and infomorphisms gives us a second duality, the classical duality of Boolean algebra. Under this duality conjunction is the dual of disjunction. For example, $\neg(\vee A) = \wedge\neg A$ and $\neg(\wedge A) = \vee\neg A$. Under this duality, unlike the flip duality, the direction of infomorphisms is not reversed.

7.2 Boolean Operations in Classifications

We can use the Boolean operations on classifications to determine what it would mean for an operation on types to be a disjunction, conjunction, or negation in a classification. Recall the notion of a token identical infomorphism from Definition 5.19.

Definition 7.5. Let A be a classification.

1. A *disjunction infomorphism* on A is a token identical infomorphism $d : \vee A \rightleftarrows A$; the corresponding operation $d^\wedge$ is called a disjunction on A. Given a disjunction infomorphism d and a set Θ of types, we often write $\vee\Theta$ for $d^\wedge(\Theta)$.
2. A *conjunction infomorphism* on A is a token identical infomorphism $c : \wedge A \rightleftarrows A$; the corresponding operation $c^\wedge$ is called a conjunction on A. Given a conjunction infomorphism c and a set Θ of types, we often write $\wedge\Theta$ for $c^\wedge(\Theta)$.
3. A *negation infomorphism* on A is a token identical infomorphism $n : \neg A \rightleftarrows A$. Given a negation infomorphism n and a type α, we often write $\neg\alpha$ for $n^\wedge(\alpha)$.

It is easy to see that if d is a token identical infomorphism, then d is a disjunction on A if and only if for every set Θ of types and every token a, $a \models_A d(\Theta)$ if and only if $a \models_A \alpha$ for some $\alpha \in \Theta$. Likewise, c is a conjunction on A if and only if for every set Θ of types and every token a, $a \models_A c(\Theta)$ if and only if $a \models_A \alpha$ for all $\alpha \in \Theta$. Similarly, n is a negation if and only if for every $\alpha \in \text{typ}(A)$ and every token a, $a \not\models_A n(\alpha)$ if and only if $a \models_A \alpha$.

7.3 Boolean Classifications

Definition 7.6. A classification B is *Boolean* if it has a disjunction $\bigvee$, conjunction $\bigwedge$, and negation $\neg$.

Boolean classifications are expressively rich. We make this precise in the next result. Call a set $X \subseteq \text{tok}(A)$ *closed under indistinguishability* in A provided that whenever $a_1 \in X$ and $a_1 \sim_A a_2$, then $a_2 \in X$. Clearly every set of

tokens of the form $X = \text{tok}(\alpha)$, for $\alpha \in \text{typ}(A)$, is closed under indistinguishability. The Boolean classifications are the classifications in which every set closed under indistinguishability has this form. We leave the proof of the following result to Exercise 7.3. An answer is provided.

Proposition 7.7. *A classification A is Boolean if and only if for every set X of tokens closed under indistinguishability there is a type α such that* $X = \text{typ}(\alpha)$.

Every classification can be canonically embedded in a Boolean classification. We define this classification here. The classification defined will turn out to be important in our discussion of state spaces.

Definition 7.8. A *partition* of a set Σ is a pair $\langle\Gamma, \Delta\rangle$ of subsets of Σ such that $\Gamma \cup \Delta = \Sigma$ and $\Gamma \cap \Delta = \emptyset$.[1] Given a classification A and a token a of A, the *state description* of a is the pair

$$\text{state}_A(a) = \langle \text{typ}(a), \text{typ}(A) - \text{typ}(a)\rangle$$

Notice that such a state description forms a partition of typ(A).

Definition 7.9. Let A be a classification. The Boolean closure of A, written *Boole*(A), is the classification defined as follows:

1. the tokens of Boole(A) are the tokens of A;
2. the types of Boole(A) are arbitrary *sets of* partitions of the types of A; and
3. the classification relation is defined by $a \models_{\text{Boole}(A)} \alpha$ if and only if $\text{state}_A(a) \in \alpha$.

We define a token identical, injective infomorphism $\eta_A : A \rightleftarrows \text{Boole}(A)$ by

$$\eta_A(\alpha) = \{\langle\Gamma, \Delta\rangle \text{ a partition} \mid \alpha \in \Gamma\}.$$

Justification. Notice that $\eta_A(\alpha)$ is indeed a set of partitions of the types of A and hence a type of our target classification. It is obvious that $\eta_A : A \rightleftarrows \text{Boole}(A)$ is an injective infomorphism. □

Proposition 7.10. *For any classification A, the operations of union, intersection, and complement are a disjunction, conjunction, and negation, respectively, on* Boole(A).

[1] These are sometimes called "quasi-partitions" because it is not assumed that the sets are non-empty. We will be using the notion too frequently to put up with such unpleasant terminology, however.

Proof. This is easy, given the definitions. For example, to see that $\bigcap$ is a conjunction, we need only note that the state description of a token is in every $\alpha \in \Gamma$ if and only if it is in $\bigcap \Gamma$. □

Exercises

7.1. Prove Proposition 7.7.

7.2. For any classification $\boldsymbol{A}$, show that
1. $\neg\neg\boldsymbol{A} = \boldsymbol{A}$
2. $\neg \wedge \boldsymbol{A} = \vee\neg\boldsymbol{A}$
3. $\neg \vee \boldsymbol{A} = \wedge\neg\boldsymbol{A}$

7.3. Show that for any classifications $\boldsymbol{A}$ and $\boldsymbol{B}$, $\neg(\boldsymbol{A} + \boldsymbol{B}) = \neg\boldsymbol{A} + \neg\boldsymbol{B}$. Conclude that $\neg(\neg\boldsymbol{A} + \neg\boldsymbol{B}) = \boldsymbol{A} + \boldsymbol{B}$.

7.4. Find an infomorphism $f : \wedge\wedge\boldsymbol{A} \rightleftarrows \wedge\boldsymbol{A}$ such that $f\eta^c_{\wedge \boldsymbol{A}} = 1_{\wedge \boldsymbol{A}}$.

7.5. Investigate the properties of the finite analogs of the Boolean operations studied in the chapter.

Lecture 8

State Spaces

State-space models are one of the most prevalent tools in science and applied mathematics. In this lecture, we show how state spaces are related to classifications and how systems of state spaces are related to information channels. As a result, we will discover that state spaces provide a rich source of information channels. In later lectures, we will exploit the relationship between state spaces and classifications in our study of local logics.

8.1 State Spaces and Projections

Definition 8.1. A *state space* is a classification $\boldsymbol{S}$ for which each token is of exactly one type. The types of a state space are called *states*, and we say that a is *in state* σ if $a \models_{\boldsymbol{S}} \sigma$. The state space $\boldsymbol{S}$ is *complete* if every state is the state of some token.

Example 8.2. In Example 4.5 we pointed out that for any function $f : A \to B$, there is a classification whose types are elements of B and whose tokens are elements of A and such that $a \models b$ if and only if $b = f(a)$. This classification is a state space and every state space arises in this way, so another way to put the definition is to say that a state space is a classification $\boldsymbol{S}$ in which the classification relation $\models_{\boldsymbol{S}}$ is a total function. For this reason, we write $\text{state}_{\boldsymbol{S}}(a)$ for the state σ of a in $\boldsymbol{S}$. In defining a state space $\boldsymbol{S}$, we will often proceed by specifying its tokens, types, and the function $\sigma = \text{state}_{\boldsymbol{A}}(a)$. The state space $\boldsymbol{S}$ is complete if and only if this function is surjective.

Because the classification relation in state spaces is a function, we depict them by diagrams of the following kind:

typ(S)

↑ $state_S$

tok(S)

We call a state space S *ideal* if typ(S) = tok(S) and for each $s \in$ tok(S), $state_S(s) = s$. For any set S there is exactly one ideal state space S with types and tokens that are the members of S. This is an extremely simple example but it lets us see that we are generalizing the usual notion of a state space, where tokens are ignored. One can simply identify the standard notion of a state space with our notion of an ideal state space. Notice that ideal state spaces are necessarily complete.

We have given a number of examples of state spaces in Lecture 3. We present some additional examples here.

Example 8.3. Let A be a sequence of n gravitating bodies, let T be some ordered set modeling time, and let

$$S = \{\langle A, t\rangle \mid t \in T\}$$

be the set of tokens, with $\langle A, t\rangle$ modeling the bodies at time t. A *Newtonian state space* with tokens S and states $\Omega = R^{6n}$ represents the position and velocity of the bodies at time t.

Example 8.4. If we depict a finite state space as a table, then there must be exactly one "1" in each row, as in the following:

S	α_1	α_2	α_3	α_4	α_5
a_1	1	0	0	0	0
a_2	1	0	0	0	0
a_3	0	1	0	0	0
a_4	0	0	1	0	0

The flip $S^{\perp}$ of a state space is not in general a state space, of course.

Example 8.5. A molecule of DNA is made up of a long double strand of simpler molecules, called *bases*, and denoted by A, T, C, and G. For example, we might have

```
G  T  A  T  C  C  G  ...
‖  |  |  |  ‖  ‖  ‖
C  A  T  A  G  G  C  ...
```

Because of the bonding properties of the bases, C and G are always paired together, as are T and A. Information about the genetic makeup of an individual plant or animal is encoded in such a strand. Strings from individuals of a single species will be similar in most regards, and quite different from individuals of other species. Because of the nature of the bonding, any strand of DNA can be characterized by a single string on the alphabet {A, T, C, G}. Thus we can form a state space S_{DNA} whose tokens are strands of DNA and whose types are finite strings on {A, T, C, G}. The state of a strand is the sequence that characterizes it.

In 1953, Crick and Watson observed that if a single strand of DNA were to split and recombine naturally with free molecules of the four bases, two strands would form, each identical in structure to the original. Following up on this idea, Meselson and Stahl showed in 1956 that such splits and recombinations do indeed account for the production of new DNA from old. These connections allow one strand to carry a great deal of information about the other because any two strands of DNA that are connected will normally (mutations being the exceptions) be of the same type. (See Exercise 8.1.)

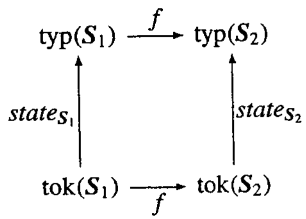
Example 8.6. In Example 4.6, we defined the truth classification of a language L. This is not a state space as every model will satisfy more than one sentence. However, there is a closely related state space where we take states to be sets of sentences and assign to each model its complete theory, the set of all sentences true in the model.

8.2 Projections and Products

Definition 8.7. A *(state space) projection* $f : S_1 \rightrightarrows S_2$ from state space S_1 to state space S_2 is given by a covariant pair of functions such that for each token $a \in$ tok(S_1),

$$f(state_{S_1}(a)) = state_{S_2}(f(a)).$$

This requirement can be rephrased by saying that the following diagram commutes:



(In this diagram we have left off the up and down symbols on f, as we often do when no confusion is likely.)

Example 8.8. Recall the Newtonian state spaces of Example 8.3. Given a sequence A of n gravitating bodies and a set T of times, we took $S = \{\langle A, t\rangle \mid t \in T\}$ as the set of tokens and $\Omega = R^{6n}$ as the set of states. A state space S with these tokens and types models the physical system through time. Now consider a subsequence A_0 of A consisting of, say, $m < n$ objects and consider S_0 to be the analogous state space representing this system of bodies in A_0 through time. There is a natural projection $f : S \rightrightarrows S_0$ defined as follows. For a token $\langle A, t\rangle$ we define $f(\langle A, t\rangle) = x\langle A_0, t\rangle$. This just assigns to the whole system at time t the subsystem at the same time. Given a state $\sigma \in R^{6n}$, $f(\sigma)$ is the vector in R^{6m} that picks out the position and velocity of the bodies in A_0.

Example 8.9. Given projections $f : S_1 \rightrightarrows S_2$ and $g : S_2 \rightrightarrows S_3$, the *composition* $gf : S_1 \rightrightarrows S_3$ of f and g is the projection defined by $(gf)^{\wedge} = g^{\wedge} f^{\wedge}$ and $(gf)^{\vee} = g^{\vee} f^{\vee}$. It is straightforward to see that gf is indeed a projection.

Example 8.10. The *identity* $1_S : S \rightrightarrows S$ on S, which is the identity on both tokens and states, is clearly a projection.

In dealing with classifications, the operation of summation was central. With state spaces, products play the analogous role.

Definition 8.11. Given state spaces S_1 and S_2, the *product* $S_1 \times S_2$ of S_1 and S_2 is the classification defined as follows:

1. The set of tokens of $S_1 \times S_2$ is the Cartesian product of tok(S_1) and tok(S_2);
2. the set of types of $S_1 \times S_2$ is the Cartesian product of typ(S_1) and typ(S_2); and
3. $\langle a, b\rangle \models_{S_1 \times S_2} \langle \alpha, \beta\rangle$ if and only if $a \models_{S_1} \alpha$ and $b \models_{S_2} \beta$.

There are natural projections $\pi_{S_1} : S_1 \times S_2 \rightrightarrows S_1$ and $\pi_{S_2} : S_1 \times S_2 \rightrightarrows S_2$ defined as follows.

1. For each $\langle \alpha, \beta\rangle \in$ typ($S_1 \times S_2$), $\pi_{S_1}(\langle \alpha, \beta\rangle) = \alpha$ and $\pi_{S_2}(\langle \alpha, \beta\rangle) = \beta$.
2. For each $\langle a, b\rangle \in$ tok($S_1 \times S_2$), $\pi_{S_1}(\langle a, b\rangle) = a$ and $\pi_{S_2}(\langle a, b\rangle) = b$.

Justification. It is clear that $S_1 \times S_2$ is a state space and that $\text{states}_{S_1 \times S_2}(\langle a, b\rangle) = \langle \text{states}_{S_1}(a), \text{states}_{S_2}(b)\rangle$. It is routine to verify that $\pi_{S_1} : S_1 \times S_2 \rightrightarrows S_1$ and $\pi_{S_2} : S_1 \times S_2 \rightrightarrows S_2$ satisfy the defining condition for projections. □

The *product* $\prod_{i \in I} S_i$ of an indexed family $\{S_i\}_{i \in I}$ of state spaces is defined similarly.

8.3 Subspaces

In order to capture the laws of a physical system, we found it convenient in Lecture 3 to carve out a subspace of a state space by means of imposing certain equations of state. Here is the general notion.

Definition 8.12. A state space S_0 is a *subspace* of a state space S_1, written $S_0 \subseteq S_1$, if the pair $\iota = \langle \iota^{\wedge}, \iota^{\vee}\rangle$ of functions that is the identity $\iota^{\wedge}$ on states and $\iota^{\vee}$ on tokens is a projection $\iota : S_0 \rightrightarrows S_1$.

Equivalently, $S_0 \subseteq S_1$ if and only if the tokens of S_0 are a subset of the tokens of S_1, the states of S_0 are a subset of the states of S_1, and the state function of S_0 is the restriction of the state function of S_1 to the tokens of S_0.

Example 8.13. Let R^{6n} be the set of states in the position and momentum state space for n bodies moving through space. The energy of the system of bodies is some (typically differentiable) function of the position and momentum of the bodies, say $E(p_1, x_1, \ldots, p_n, x_n)$. (Here p_i is the triple representing the position of the ith body, x_i its momentum.) If the system is isolated, then its energy remains constant over time, at some value, say e_0. Thus the states the system can actually occupy are the solutions to the equation

$$E(p_1, x_1, \ldots, p_n, x_n) = e_0.$$

If E is differentiable, the solution space is a differential manifold M embedded in R^{6n} and the system will always have a state in M. Restricting the tokens to those whose states are in M will capture the fact that energy is conserved in isolated physical systems.

Definition 8.14. Let $f : S \rightrightarrows S'$ be a state space projection.

1. Given a subspace S_0 of S, the *image* of S_0 under f, written $f[S_0]$, is the subspace of S' whose tokens are those in the range of $f^{\vee}$ and whose states are those in the range of $f^{\wedge}$.
2. Similarly, given a subspace S_1 of S', the *inverse image* of S_1 under f, written $f^{-1}[S_1]$, is the subspace of S whose tokens are those in $f^{\vee-1}[\mathrm{tok}(S_1)]$ and whose states are those in $f^{\wedge-1}[\mathrm{typ}(S_1)]$.

Justification. We need to verify that $f[S_0]$ is a subspace of S' and that $f^{-1}[S_1]$ is a subspace of S. We verify the latter and leave the former to the reader. Suppose a is a token of $f^{-1}[S_1]$. We need to make sure that $\mathrm{state}_S(a)$ is a state in $f^{-1}[S_1]$. By definition, $f(a)$ is a token of S_1. Hence $\mathrm{state}_{S'}(f(a))$ is a state of S_1, because $S_1 \subseteq S'$. But $\mathrm{state}_{S'}(f(a)) = f(\mathrm{state}_S(a))$ because f is a projection. Hence $\mathrm{state}_S(a)$ is a state in $f^{-1}[S_1]$, as desired. □

8.4 Event Classifications

Given a state space S, its states represent total information about the tokens, relative to the classification. Usually, however, we have only partial information about a token. The common way around this is to turn to the "event classification" of S, which we write as Evt(S). (The terminology, not entirely felicitous for reasons explained in Lecture 2, comes from probability theory.) Projections on state spaces correspond to infomorphisms of the associated event classifications.

Definition 8.15. The *event classification* Evt(S) associated with a state space S has as tokens the tokens of S. Its types are arbitrary sets of states of S. The classification relation is given by $a \vDash_{\mathrm{Evt}(S)} \alpha$ if and only if $\mathrm{state}_S(a) \in \alpha$.

As we saw in Lecture 3, these event classifications are a useful tool for understanding the relationship between linguistic accounts of the world, both scientific and common sense accounts, and scientific models of the world.

Example 8.16. Recall the DNA state space S_{DNA} from Example 8.5. The classification Evt(S_{DNA}) is quite useful. Certain sets of states correspond to genetic traits. For example, there will be certain sets of states that correspond to DNA strands from humans with green eyes. This suggests the following infomorphism. Let ***Genes*** be a classification of humans by their genetic traits. Thus the types of ***A*** include things like the gene GREEN-EYES for green eyes, and so forth, with $a \vDash_{A} \alpha$ if and only if a carries the genetic trait α. Let *hgp*: ***Gene*** $\rightleftarrows$ Evt(S_{DNA}) be defined on (gene) types by the following: *hgp*(α) is the set of strings that characterize strands of DNA of individuals of type α. On tokens, *hgp*(s) is the individual whose strand of DNA is s. This is an infomorphism because genetic traits are entirely determined by DNA. The famous Human Genome Project can be thought of as attempting to characterize this infomorphism, which is why we have dubbed it "*hgp*."

Let S_1 and S_2 be state spaces and let $f : S_1 \rightrightarrows S_2$ be covariant. Define a contravariant pair $\mathrm{Evt}(f) : \mathrm{Evt}(S_2) \rightleftarrows \mathrm{Evt}(S_1)$ as follows. On tokens, Evt(f) and f are the same: $\mathrm{Evt}(f)^{\vee}(a) = f^{\vee}(a)$ for $a \in \mathrm{tok}(\mathrm{Evt}(S_1))$. On types $\alpha \in \mathrm{typ}(\mathrm{Evt}(S_2))$, $\mathrm{Evt}(f)(\alpha) = f^{-1}[\alpha]$.

Proposition 8.17. *Given state spaces S_1 and S_2, the following are equivalent:*

1. $f : S_1 \rightrightarrows S_2$ *is a projection;*
2. $\mathrm{Evt}(f) : \mathrm{Evt}(S_2) \rightleftarrows \mathrm{Evt}(S_1)$ *is an infomorphism.*

Proof. Assume $f : S_1 \rightrightarrows S_2$ is a state-space projection. To show that Evt(f) is an infomorphism, let $a \in \mathrm{tok}(\mathrm{Evt}(S_1))$ and let $\alpha \in \mathrm{typ}(\mathrm{Evt}(S_2))$. Then

$$
\begin{aligned}
\mathrm{Evt}(f)(a) \vDash_{\mathrm{Evt}(S_2)} \alpha \quad &\text{iff} \quad \mathrm{state}_{S_2}(f(a)) \in \alpha \\
&\text{iff} \quad f(\mathrm{state}_{S_1}(a)) \in \alpha \\
&\text{iff} \quad \mathrm{state}_{S_1}(a) \in f^{-1}[\alpha] \\
&\text{iff} \quad a \vDash_{\mathrm{Evt}(S_1)} f^{-1}[\alpha] \\
&\text{iff} \quad a \vDash_{\mathrm{Evt}(S_1)} \mathrm{Evt}(f)(\alpha)
\end{aligned}
$$

To prove the converse, suppose Evt(f) is an infomorphism. We need to check that $\mathrm{state}_{S_2}(f(a)) = f(\mathrm{state}_{S_1}(a))$. Let $\mathrm{state}_{S_1}(a) = \sigma$. We show that $\mathrm{state}_{S_2}(f(a)) = f(\sigma)$. Let $\sigma' = \mathrm{state}_{S_2}(f(a))$, which we can write as $f(a) \vDash_{\mathrm{Evt}(S_2)} \{\sigma'\}$. Because Evt($f$) is an infomorphism,

$$f(a) \vDash_{\mathrm{Evt}(S_2)} \{\sigma'\} \quad \text{iff} \quad a \vDash_{\mathrm{Evt}(S_1)} f^{-1}(\sigma'),$$

so $a \vDash_{\mathrm{Evt}(S_1)} f^{-1}(\sigma')$, which means that $\sigma \in f^{-1}(\sigma')$, that is, $f(\sigma) = \sigma'$, as desired. □

The following makes event classifications especially well behaved.

Proposition 8.18. *For any state space S, the classification* Evt(S) *is a Boolean classification. Indeed, the operations of intersection, union, and complement are a conjunction, disjunction, and negation, respectively.*

Proof. These are all routine. Let us prove that the complement operation is a negation. For readability, we write $-\alpha$ for typ(*A*) $-\alpha$. We show that for any set α of states, $a \not\models_{\mathrm{Evt}(S)} \alpha$ if and only if $a \models_{\mathrm{Evt}(S)} -\alpha$. But $a \not\models_{\mathrm{Evt}(S)} \alpha$ if and only if $\mathrm{state}_S(a) \notin \alpha$ if and only if $\mathrm{state}_S(a) \in -\alpha$ if and only if $a \models_{\mathrm{Evt}(S)} -\alpha$, as desired. □

8.5 State Spaces from Classifications

The classifications naturally associated with first-order logic, where tokens are structures and sentences are types, are not state spaces. We saw in Example 8.6, however, that there is a natural state space associated with each such classification, namely, where we take the state of a structure to be its complete theory. We can do something similar for an arbitrary classification.

Given a classification A and a token a of A, recall from Definition 7.8 that the *state description* of a is the pair

$$\mathrm{state}_A(a) = \langle \mathrm{typ}(a), \mathrm{typ}(A) - \mathrm{typ}(a) \rangle.$$

Thus the state description forms a partition of typ(A).

Definition 8.19. Given a classification A, the *free state space* Ssp(A) of A is defined as follows:

1. The tokens of Ssp(A) are the same as those of A;
2. the states of Ssp(A) are arbitrary partitions $\langle \Gamma, \Delta \rangle$ of the types of A; and
3. $\mathrm{state}_{\mathrm{Ssp}(A)}(a)$ is the state description $\mathrm{state}_A(a)$ of a in A.

Let A and B be classifications, and let $f : A \rightleftarrows B$ be a contravariant pair of functions. Define a covariant pair of functions $\mathrm{Ssp}(f) : \mathrm{Ssp}(B) \rightrightarrows \mathrm{Ssp}(A)$ as follows. On tokens, Ssp(f) and f are the same: $\mathrm{Ssp}(f)^{\vee}(b) = f^{\vee}(b)$ for $b \in \mathrm{tok}(B)$. On partitions, Ssp(f) is defined by

$$\mathrm{Ssp}(f)(\langle \Gamma, \Delta \rangle) = \langle f^{\wedge -1}[\Gamma], f^{\wedge -1}[\Delta] \rangle.$$

Proposition 8.20. *Let A and B be classifications, and let $f : A \rightleftarrows B$ be a contravariant pair of functions. The following are equivalent:*

1. *$f : A \rightleftarrows B$ is an infomorphism;*
2. *$\mathrm{Ssp}(f) : \mathrm{Ssp}(B) \rightrightarrows \mathrm{Ssp}(A)$ is a state space projection.*

Proof. To prove the implication from (1) to (2), we need to show that for all $b \in \mathrm{tok}(B)$, $\mathrm{Ssp}(f)(\mathrm{state}_B(b)) = \mathrm{state}_A(\mathrm{Ssp}(f)(b))$. To simplify notation, let $g = \mathrm{Ssp}(f)$ and let $\mathrm{state}_B(b) = \langle \Gamma, \Delta \rangle$ be the state description of b in B. To show that $g(\langle \Gamma, \Delta \rangle)$ is the state description of $g(b) = f(b)$ in A, we need to show that $\alpha \in f^{\wedge -1}[\Gamma]$ if and only if $f(b) \models_A \alpha$ and similarly that $\alpha \in f^{\wedge -1}[\Delta]$ if and only if $f(b) \not\models_A \alpha$ for every $\alpha \in \mathrm{typ}(A)$. The first of these is verified as follows:

$$\begin{aligned} \alpha \in f^{\wedge -1}[\Gamma] \quad &\text{iff} \quad f(\alpha) \in \Gamma \\ &\text{iff} \quad b \models_B f(\alpha) \\ &\text{iff} \quad f(b) \models_A \alpha. \end{aligned}$$

The second verification is similar.

To prove that (2) implies (1), assume that Ssp(f) is a state-space projection. We need to check that for $b \in \mathrm{tok}(B)$ and $\alpha \in \mathrm{typ}(A)$, $f(b) \models_A \alpha$ if and only if $b \models_B f(\alpha)$. Suppose that $b \not\models_B f(\alpha)$. Then $f(\alpha)$ is in the second coordinate of the state description of b in B. But then because Ssp(f) is a projection, α is in the second coordinate of the state description of $f(b)$ in A. But that means $f(b) \not\models_A \alpha$. The proof of the converse is similar, using first coordinates. □

8.6 Back to Boole

The question naturally arises as to what happens if we start with a state space, construct its event classification, and then construct the state space associated with that. Or, to start in the other way, we can begin with a classification, construct its state space, and then construct its event classification. In this section we study this question.

In Lecture 3, we relied at one point on the fact that any infomorphism $f : A \rightleftarrows \mathrm{Evt}(S)$ of a classification into an event classification has a natural extension to the Boolean closure Boole(A) of A; that is, there is an f^* agreeing with f of A such that $f^* : \mathrm{Boole}(A) \rightleftarrows \mathrm{Evt}(S)$ is an infomorphism. The reason for this will fall out of our discussion.

Proposition 8.21. *For any classification A,*

$$\mathrm{Boole}(A) = \mathrm{Evt}(\mathrm{Ssp}(A)).$$

Proof. This amounts to unwinding the two definitions to see that they come to the same thing. □

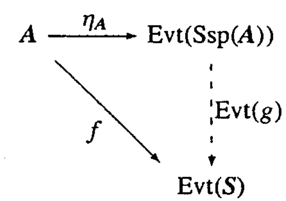
The correspondence between state spaces and event classifications is closely related to the Stone representation theorem for Boolean algebras. (If we define the obvious Boolean algebra on the type set of a Boolean classification then the partitions used in the construction of its state-space are in one-to-one correspondence with the points of the corresponding Stone space. Our main addition to the Stone representation theorem is in keeping track of the classification relation throughout the construction. See Davey and Priestlay (1990) for an excellent introduction to representation theorems.)

Recall the token identical embedding of $\eta_A : A \rightleftarrows \text{Boole}(A)$ of a classification A into its Boolean closure. In view of the above proposition, this means we have a token identical embedding $\eta_A : A \rightleftarrows \text{Evt}(\text{Ssp}(A))$. On types $\alpha \in \text{typ}(A)$

$$\eta_A(\alpha) = \{\langle \Gamma, \Delta \rangle \text{ a partition} \mid \alpha \in \Gamma\}.$$

In mathematical terms, the next result shows that for any classification A, the state space $\text{Ssp}(A)$ is "free" on A. It can also be taken as showing that given a state space S and a classification A, there is a natural bijection between infomorphisms $f : A \rightleftarrows \text{Evt}(S)$ and projections $g : S \rightrightarrows \text{Ssp}(A)$.

Proposition 8.22. *Let S be a state space and let A be a classification. For every infomorphism $f : A \rightleftarrows \text{Evt}(S)$ there is a unique projection $g : S \rightrightarrows \text{Ssp}(A)$ such that the following diagram commutes:*



Proof. First, assume $g : S \rightrightarrows \text{Ssp}(A)$ is any projection that makes the diagram commute. Because η_A is the identity on tokens, it is clear that f and g agree on tokens. On types, the commuting of the diagram insures that

$$f(\alpha) = \{\sigma \in \text{typ}(S) \mid \alpha \in 1^{st}(g(\sigma))\}.$$

But this means that

$$\sigma \in f(\alpha) \quad \text{iff} \quad \alpha \in 1^{st}(g(\sigma)).$$

Because $g : S \rightrightarrows \text{Ssp}(A)$, this tells us that

$$g\hat{}(\sigma) = \langle \{\alpha \mid \sigma \in f(\alpha)\}, \{\alpha \mid \sigma \notin f(\alpha)\} \rangle.$$

So let us use this equation to define $g\hat{}$. It is easy to check that g is a projection and that the diagram commutes. □

This proposition expresses a much simpler idea than it might seem, as is shown by the following example.

Example 8.23. Recall the infomorphism hgp: ***Gene*** $\rightleftarrows \text{Evt}(S_{DNA})$ encoding genetic traits of humans in terms of strands of their DNA. This infomorphism has the form of f in the proposition, so we are promised a projection $g : S_{DNA} \rightrightarrows \text{Ssp}(\textbf{\textit{Gene}})$ bearing a certain relationship to f. This projection is the same as hgp on tokens; that is, it assigns to each strand of DNA the individual from which it came. On a state $\sigma \in \{\text{A}, \text{T}, \text{C}, \text{G}\}^*$, g partitions the genetic traits into those that are compatible with DNA of type σ and those that are not. The infomorphism hgp and the projection g are, up to the embedding of ***Gene*** in its Boolean closure, two ways of looking at the same coding of human genetic traits by DNA.

As a consequence of the proposition, we obtain the desired ability to lift infomorphisms to Boolean closures of classifications.

Corollary 8.24. *Every infomorphism f of the form $f : A \rightleftarrows \text{Evt}(S)$ of a classification into an event classification has a natural extension to an infomorphism $f^* : \text{Boole}(A) \rightleftarrows \text{Evt}(S)$.*

Proof. In view of Proposition 8.21, we can just let $f^* = \text{Evt}(g)$, where g is given as in Proposition 8.22. □

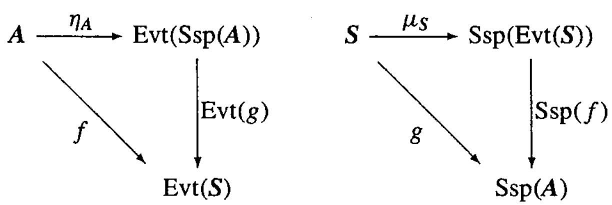
If we examine the proof of Proposition 8.22, we can get a little more information from it. Namely, we can see how to define the projection g from the infomorphism f. First we need a definition.

Definition 8.25. For any state space S, let $\mu_S : S \rightrightarrows \text{Ssp}(\text{Evt}(S))$ be the injective, token-identical projection defined as follows: for any state $\sigma \in \text{typ}(S)$

$$\mu_S(\sigma) = \langle \{\Gamma \subseteq \text{typ}(S) \mid \sigma \in \Gamma\}, \{\Delta \subseteq \text{typ}(S) \mid \sigma \notin \Delta\} \rangle.$$

Justification. Notice that $\mu_S(\sigma)$ is indeed a partition of the types of $\text{Evt}(S)$ and hence a state of our target state space. It is obvious that $\mu_S : S \rightrightarrows \text{Ssp}(\text{Evt}(S))$ is an injective projection. □

Corollary 8.26. *If the diagram on the left commutes, then so does that on the right.*



Proof. Recall that Ssp(f) agrees with f on tokens and on types satisfies

$$\text{Ssp}(f)(\langle\Gamma,\Delta\rangle) = \langle f^{\wedge-1}[\Gamma], f^{\wedge-1}[\Delta]\rangle.$$

On the other hand,

$$\mu_S(\sigma) = \langle\{\Gamma \subseteq \text{typ}(S) \mid \sigma \in \Gamma\}, \{\Delta \subseteq \text{typ}(S) \mid \sigma \notin \Delta\}\rangle.$$

Thus

$$\text{Ssp}(f)(\mu_S(\alpha)) = \langle\{\alpha \mid \sigma \in f(\alpha)\}, \{\alpha \mid \sigma \notin f(\alpha)\}\rangle.$$

Thus to say that the diagram commutes is just to say that f and g agree on tokens and that

$$g(\sigma) = \langle\{\alpha \mid \sigma \in f(\alpha)\}, \{\alpha \mid \sigma \notin f(\alpha)\}\rangle$$

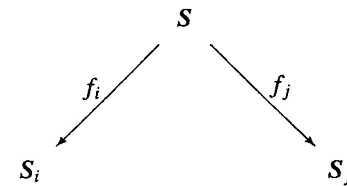
But this is the definition of g in the proof of Proposition 8.22. □

8.7 State-Space Systems

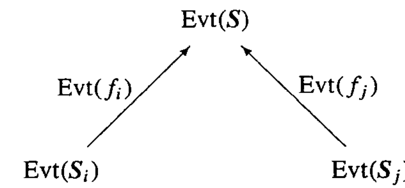
The state-space model of a complex system consists of a state space for the system as a whole, together with projections into state spaces for each component. Thus we give the following definition.

Definition 8.27. A *state-space system* consists an indexed family $\mathcal{S} = \{f_i : S \rightrightarrows S_i\}_{i\in I}$ of state-space projections with a common domain S, called the *core* of $\mathcal{S}$, to state spaces S_i (for $i \in I$); S_i is called the ith component space of $\mathcal{S}$.

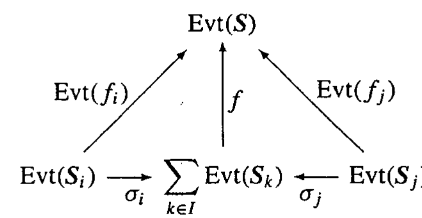
We can transform any state-space system into an information channel by applying the functor Evt. This is particularly obvious if we use diagrams. The system $\mathcal{S}$ consists of a family of projections, of which we depict two:



Applying the operator Evt to this diagram gives a family of infomorphisms with a common codomain, Evt(S):



This is an information channel. Using the fundamental property of sums, we get the following commuting diagram, where for the sake of readability, we write σ_i for $\sigma_{\text{Evt}(S_i)}$, and f for $\sum_{k\in I}\text{Evt}(f_k)$:



This diagram will be quite useful to us in Lecture 15, once we learn how to move logics along infomorphisms.

We could also go the other way around, turning any channel into a state-space system by means of the functor Ssp.

Exercises

8.1. Construct an information channel representing information flow between two strands of DNA that have the same parent, that is, that arose from the normal splitting of a single strand of DNA.

8.2. Show that state spaces S_1 and S_2 are isomorphic (as classifications) if and only if there is a projection $f : S_1 \rightrightarrows S_2$ such that both $f^\wedge$ and $f^\vee$ are bijections.

8.3. Show that if state spaces S_1 and S_2 are both complete, so is their product.

8.4. For any state space S, we define the *idealization* of S to be the state space with types and tokens that are the set of types of S. Show that there is a projection f from S to its idealization. This shows that every state space is the inverse image of an ideal state space under some projection. Note that ideal state spaces are complete, so that we also find that every state space is the inverse image of a complete state space under some projection.

8.5. Show that for any classifications A and B,

$$\mathrm{Ssp}(A+B) \cong \mathrm{Ssp}(A) \times \mathrm{Ssp}(B).$$

(This result is a consequence of the adjointness results proven above, together with well-known facts about adjoint functors. However, for most of us, it is easier to prove directly.)

8.6. (†) Verify that Evt()is a functor from the category of state spaces (with projections as morphisms) to classifications (with infomorphisms) by showing the following:
1. If $f : S_1 \rightrightarrows S_2$ and $g : S_2 \rightrightarrows S_3$ are projections of state spaces, then $\mathrm{Evt}(gf) = \mathrm{Evt}(f)\mathrm{Evt}(g)$.
2. If S is any state space, then $\mathrm{Evt}(1_S) = 1_{\mathrm{Evt}(S)}$.

8.7. (†) Verify that Ssp() is functor from classifications to state spaces by proving the following:
1. If $f : A \rightleftarrows B$ and $g : B \rightleftarrows C$ are infomorphisms, then $\mathrm{Ssp}(gf) = \mathrm{Ssp}(f)\mathrm{Ssp}(g)$.
2. If A is any classification, then $\mathrm{Ssp}(1_A) = 1_{\mathrm{Ssp}(A)}$.

Lecture 9

Regular Theories

In this lecture, we prepare the way for the notion of local logic by studying the ways that classifications give rise to "regular theories." These theories can be seen as an idealized version of the scientific laws supported by a given closed system. The adjective "regular" refers to the purely structural properties that any such theory must satisfy. Any theory with these properties can be obtained from a suitable classification. At the end of the lecture, we will return to the question of how different scientific theories, based on different models of the phenomena under study, can be seen as part of a common theory. We will see conditions under which this obtains.

9.1 Theories

One way to think about information flow in a distributed system is in terms of a "theory" of the system, that is, a set of known laws that describe the system. Usually, these laws are expressed in terms of a set of equations or sentences of some scientific language. In our framework, these expressions are modeled as the types of some classification. However, we will not model a theory by means of a set of types. Because we are not assuming that our types are closed under the Boolean operations, as they are not in many examples, we get a more adequate notion of theory by following Gentzen and using the notion of a sequent.

Given a set Σ, a *sequent* of Σ is a pair $\langle \Gamma, \Delta \rangle$ of subsets of Σ. $\langle \Gamma, \Delta \rangle$ is *finite* if $\Gamma \cup \Delta$ is a finite set. Recall that a sequent $\langle \Gamma, \Delta \rangle$ is a *partition* of a set Σ' if $\Gamma \cup \Delta = \Sigma'$ and $\Gamma \cap \Delta = \emptyset$.

Definition 9.1. A binary relation $\vdash$ between subsets of Σ is called a *(Gentzen) consequence relation on* Σ. A *theory* is a pair $T = \langle \Sigma, \vdash \rangle$, where $\vdash$ is a consequence relation on Σ. A *constraint* of the theory T is a sequent $\langle \Gamma, \Delta \rangle$ of Σ for which, $\Gamma \vdash \Delta$.

We will use the normal notational conventions about consequence relations. For example, we write $\alpha, \beta \vdash \gamma$ for $\{\alpha, \beta\} \vdash \{\gamma\}$ and $\Gamma, \Gamma' \vdash \Delta, \alpha$ for $\Gamma \cup \Gamma' \vdash \Delta \cup \{\alpha\}$. We have already given several examples of constraints in Lecture 2. Here is another.

Example 9.2. Suppose that Σ is the set of polynomials in the variables x and y, and let $\vdash$ be the consequence relation consisting of sequents $\langle \Gamma, \Delta \rangle$ such that every pair $\langle r_1, r_2 \rangle \in R^2$ of real numbers satisfying all the equations in Γ satisfies some equation in Δ. For example, one constraint of the resulting theory is

$$x^2 + y^2 = 25, 3x = 4y \vdash x = 4, x = -4.$$

The point of this example is twofold: to illustrate the way sequents allow us to express certain constraints that would not be expressible without otherwise bringing in logical operations on types, and to stress that the comma on the left has a conjunctive force, whereas that on the right has a disjunctive force.

Example 9.3. Given a first-order language L, recall the truth classification of L given in Example 4.6 whose types are sentences of L and whose tokens are L-structures. The theory of this classification has as constraints just the sequents of first-order logic that are valid in the usual sense. For example, the sequent

$$\forall x\, [A(x) \rightarrow B(x)] \vdash \neg A(c), B(c)$$

is one such constraint.

Our definition of a theory is intended to be neutral between semantic and proof-theoretic notions of theory. Our primary examples of theories come from classifications and from state spaces.

Definition 9.4. Given a classification A, a token $a \in \text{tok}(A)$ *satisfies* a sequent $\langle \Gamma, \Delta \rangle$ of $\text{typ}(A)$ provided that if a is of every type in Γ, then it is of some type in Δ. A token not satisfying a sequent is called a *counterexample* to the sequent. The theory $\text{Th}(A) = \langle \text{typ}(A), \vdash_A \rangle$ *generated by* a classification A is the theory whose types are the types of A and whose constraints are the set of sequents satisfied by every token in A.

Proposition 9.5. *The theory* $\text{Th}(A) = \langle \text{typ}(A), \vdash_A \rangle$ *generated by a classification A satisfies the following for all types* α *and all sets* $\Gamma, \Gamma', \Delta, \Delta', \Sigma', \Sigma_0, \Sigma_1$ *of types:*

Identity: $\alpha \vdash \alpha$.
Weakening: *If* $\Gamma \vdash \Delta$, *then* $\Gamma, \Gamma' \vdash \Delta, \Delta'$.
Global Cut: *If* $\Gamma, \Sigma_0 \vdash \Delta, \Sigma_1$ *for each partition* $\langle \Sigma_0, \Sigma_1 \rangle$ *of* Σ', *then* $\Gamma \vdash \Delta$.

Proof. It is clear that $\vdash_A$ satisfies Identity and Weakening. Let us show that it satisfies Global Cut. Suppose a is a counterexample to $\langle \Gamma, \Delta \rangle$ and that $\Sigma' \subseteq \text{typ}(A)$. Let $\Sigma_0 = \{\alpha \in \Sigma' \mid a \vDash \alpha\}$, and let $\Sigma_1 = \{\alpha \in \Sigma' \mid a \nvDash \alpha\}$. This gives us a partition of Σ'. Clearly, a is also a counterexample to $\langle \Gamma \cup \Sigma_0, \Delta \cup \Sigma_1 \rangle$. □

We generalize these properties to arbitrary theories.

Definition 9.6. A theory $T = \langle \Sigma, \vdash \rangle$ is *regular* if it satisfies Identity, Weakening, and Global Cut.

It is perhaps worth noting that there are some other structural rules present in some treatments of logic via Gentzen sequents that are made unnecessary by our decision to treat sequents as pairs of sets. For example, the familiar rules of permutation and contraction are not needed, as illustrated by the fact that $\{\alpha, \beta\} = \{\beta, \beta, \alpha\}$. If we were dealing with pairs of sequences, rather than pairs of sets, we would need additional closure conditions in the notion of regular theory.

Readers familiar with the rule of Cut in logic will find our use of this term nonstandard.[1] We call the usual Cut rule "Finite Cut." We will state it in a moment and show it is a consequence of Global Cut and, under certain conditions, is equivalent to it. It is convenient to define the following partial order on sequents of Σ:

$$\langle \Gamma, \Delta \rangle \leq \langle \Gamma', \Delta' \rangle \quad \text{iff} \quad \Gamma \subseteq \Gamma' \quad \text{and} \quad \Delta \subseteq \Delta'.$$

If $\langle \Gamma, \Delta \rangle \leq \langle \Gamma', \Delta' \rangle$, then we say that $\langle \Gamma', \Delta' \rangle$ *extends*, or is an *extension* of, the sequent $\langle \Gamma, \Delta \rangle$.

[1] We found Global Cut in the course of characterizing the theories of classifications, only to discover out that it had been previously studied in M. Dunn and G. Hardegree (unpublished manuscript, 1993). The terminology Global Cut is borrowed from their work.

Proposition 9.7. *For every theory* $T = \langle \Sigma, \vdash \rangle$, *there is a smallest regular theory on* Σ *containing the sequents in* $\vdash$ *as constraints. This is called the* regular closure *of* T.

Proof. If we take as a theory all sequents on Σ, it is regular, that is, it satisfies Identity, Weakening, and Global Cut. Now consider the intersection of all regular consequence relations containing $\vdash$. It is easy to see that it too is a regular consequence relation, simply from the general form of the definition. □

Proposition 9.8. *Any regular theory* $T = \langle \Sigma, \vdash \rangle$ *satisfies the following conditions:*

Finite Cut: *If* $\Gamma, \alpha \vdash \Delta$ *and* $\Gamma \vdash \Delta, \alpha$ *then* $\Gamma \vdash \Delta$.
Partition: *If* $\Gamma' \vdash \Delta'$ *for each partition of* Σ *with* $\langle \Gamma', \Delta' \rangle \geq \langle \Gamma, \Delta \rangle$, *then* $\Gamma \vdash \Delta$.

Proof. Both conditions are special cases of Global Cut, the first with $\Sigma' = \{\alpha\}$, the second with $\Sigma' = \Sigma$. □

The form of the Finite Cut rule makes it clear why this is called Cut: The type α that appears in both premises is cut out of the conclusion. In practice, we typically use one or the other of these special cases of Global Cut. In the presence of Weakening, Global Cut is equivalent to Partition. We will use this fact repeatedly, so we prove it here.

Proposition 9.9. *Let* $T = \langle \Sigma, \vdash \rangle$ *be any theory. If* T *satisfies Weakening and Partition, then* T *is regular.*

Proof. We need to verify Identity and Global Cut. Identity follows from Partition simply because there are no partitions extending $\langle \{\alpha\}, \{\alpha\} \rangle$. To verify Global Cut, assume the premise of Global Cut; that is, $\Gamma, \Sigma_0 \vdash \Delta, \Sigma_1$ for each partition $\langle \Sigma_0, \Sigma_1 \rangle$ of Σ'. In order to apply Partition, we need only show that $\Gamma, \Sigma_0 \vdash \Delta, \Sigma_1$ for each partition $\langle \Sigma_0, \Sigma_1 \rangle$ of Σ. But any such partition of Σ is an extension of some partition of Σ', so we obtain the result by Weakening. □

The following definition is going to seem very odd if the sequents mentioned are thought of in terms of constraints. But sequents are not only used in stating constraints, things that must happen, they are also good for talking about what might happen. This definition should be understood in this sense.

Definition 9.10. Given a regular theory $T = \langle \Sigma, \vdash \rangle$, a sequent $\langle \Gamma, \Delta \rangle$ is *T-consistent* if $\Gamma \nvdash \Delta$. The theory T is consistent if it has a consistent sequent.

Example 9.11. If A is a classification, then a sequent of the form $\langle \Gamma, \emptyset \rangle$ is $\mathrm{Th}(A)$-consistent if and only if there is a token $a \in \mathrm{tok}(A)$ such that $a \models_A \alpha$ for all $\alpha \in \Gamma$.

Intuitively, $\langle \Gamma, \Delta \rangle$ is consistent if, as far as the theory T knows, it is possible that everything in Γ could hold but, simultaneously everything in Δ could fail. This is clearly the case if our theory is the theory of a classification. If a theory T is fixed, we will omit the "T" and just call a sequent consistent. The following states for the record some obvious equivalences.

Proposition 9.12. *Given a classification* A *and a sequent* $\langle \Gamma, \Delta \rangle$ *of* $\mathrm{typ}(A)$, *the following are equivalent:*

1. $\langle \Gamma, \Delta \rangle$ *is* $\mathrm{Th}(A)$*-consistent.*
2. $\langle \Gamma, \Delta \rangle$ *is a subsequent of the state description of some* a *in* A.
3. *There is a token* a *that is a counterexample to* $\langle \Gamma, \Delta \rangle$.

Hence $\mathrm{Th}(A)$ *is consistent if and only if* $\mathrm{tok}(A) \neq \emptyset$.

It is sometimes convenient to specify a theory by specifying the set of its consistent partitions. This is a legitimate way to proceed, by virtue of Proposition 9.14. But first we need the following observation.

Proposition 9.13. *Every regular theory satisfies the following condition:*

$\Gamma \vdash \Delta$ *if and only if there is no consistent partition extending* $\langle \Gamma, \Delta \rangle$.

Conversely, any theory satisfying this condition is regular.

Proof. The first claim follows from Weakening in one direction and Partition in the other. As for the second claim, it is easy to check that any theory satisfying the condition satisfies Identity, Weakening, and Partition. For example, to see that $\alpha \vdash \alpha$, suppose it does not. Then by the condition there is a consistent *partition* extending $\langle \{\alpha\}, \{\alpha\} \rangle$. But there can be no such partition. The other rules are checked similarly. □

We can now prove the result that justifies defining a theory by giving a set of consistent partitions.

Proposition 9.14. *Every set P of partitions of Σ is the set of consistent partitions of a unique regular theory on Σ.*

Proof. Define

$\Gamma \vdash \Delta$ if and only if there is no partition in P extending $\langle \Gamma, \Delta \rangle$.

By Proposition 9.13, this definition defines a regular theory. But any regular theory with this relation as its set of consistent partitions must also satisfy the above biconditional, so there is only one. □

Example 9.15. Let $\Sigma = \{\alpha, \beta, \gamma\}$. Think of these types as atomic propositions and construct the truth table that displays all possible assignments of truth values to these three propositions. As with classification tables we use "1" and "0" for truth and falsity, respectively.

α	β	γ
1	1	1
1	1	0
1	0	1
1	0	0
0	1	1
0	1	0
0	0	1
0	0	0

Each row of this truth table corresponds to a partition $\langle \Gamma, \Delta \rangle$ of Σ: put a type into Γ if the type has a 1 under it, and into Δ if it has an 0 under it. Conversely, every partition of Σ arises in this way. Thus giving a set of partitions is nothing more than giving a set of rows of this truth table, those that are possible distributions of truth values among the atomic types.

The following notion will be useful when we turn to characterizing the logical operations on theories.

Definition 9.16. Let A be a classification and $\Sigma \subseteq \text{typ}(A)$. A partition $\langle \Gamma, \Delta \rangle$ of Σ is *realized in* A if there is a token $a \in \text{tok}(A)$ such that

$$\Gamma = \{\alpha \in \Sigma \mid a \models_A \alpha\}.$$

Otherwise $\langle \Gamma, \Delta \rangle$ is said to be *spurious in* A. The set Σ is *independent in* A if every partition of Σ is realized in A.

If $\Sigma = \text{typ}(A)$, then a partition $\langle \Gamma, \Delta \rangle$ of Σ is realized in A if it is the state description of some token $a \in \text{tok}(A)$. However, we also want to apply the notions in cases where Σ is a proper subset of typ(A).

Example 9.17. Suppose we are classifying light bulbs and have among our types the set Σ consisting of LIT, UNLIT, and LIVE. In the everyday classification of bulbs using these types, the types are far from independent. The partitions normally realized correspond to the rows of the following table:

Lit	Unlit	Live
1	0	1
0	1	1
0	1	0

The other partitions are normally spurious. Hence the intuitive theory of light bulbs, at least as far as Σ is concerned, can be specified by the consistent partitions represented by the rows of this truth table.

We can use these ideas to help us compute the theory of a classification A from its classification table. In this way we give substance to the intuition that the regularities of a classification correspond to patterns in the classification table of A. First, note that each row of such a table corresponds to a consistent partition of the types. Hence each such row must satisfy every constraint of T in the following sense: if $\Gamma \vdash_T \Delta$ and every element of Γ has a 1 under it in the given row, then some element of Δ has a 1 under it in the same row. Each missing row corresponds to a spurious partition in the corresponding theory and will thereby falsify some constraint of the theory. We codify this simple but useful observation as follows.

Proposition 9.18. *Let A be a classification and T be a theory on* typ(A). *The regular closure of T is* Th(A) *if and only if every row of the classification table of A satisfies each constraint of T and each missing row falsifies some constraint of T.*

Proof. Assume that the regular closure of T is Th(A). Each token of A must satisfy each constraint of T, so clearly every row of the classification table satisfies each sequent of T. Suppose we have a missing row. This corresponds to an inconsistent partition of typ(A), that is, a partition $\langle \Gamma, \Delta \rangle$ such that $\Gamma \vdash_A \Delta$. But then the row in question falsifies this constraint of T.

For the converse, assume that every row of the classification table of A satisfies each constraint of T and each missing row falsifies some constraint of T. We want to show that the regular closure of T is Th(A). Because every row of the classification table of A satisfies each constraint of T, and because Identity, Weakening, and Cut are sound, every row of the classification table of A satisfies each constraint of the regular closure of T. Hence every constraint of the regular closure is a constraint of Th(A). To show they are the same, suppose we have a consistent sequent $\Gamma \nvdash_T \Delta$ of the regular closure. By Partition, we can assume that this is a partition. But every partition corresponds to a possible row of the classification table. This row cannot be missing, otherwise it would falsify some constraint of T. Hence the row must correspond to some token of A. Hence $\Gamma \nvdash_A \Delta$. □

Example 9.19. We use Proposition 9.18 to show that the theory defined by the truth table in Example 9.17 is the least regular theory such that

$$\text{LIT} \vdash \text{LIVE}, \quad \vdash \text{LIT, UNLIT}, \quad \text{LIT, UNLIT} \vdash .$$

Let A be the classification whose types are those of Example 9.17 and whose tokens are the rows of the truth table, with the natural classification relation. The constraints are satisfied by each of these rows and each missing row falsifies at least one of the constraints.

Exercises

9.1. Let T be a regular theory. Show that T is inconsistent if and only if $\emptyset \vdash_T \emptyset$.

9.2. For any theory T, define an ordering on its types by $\alpha \leq_T \beta$ if and only if $\alpha \vdash_T \beta$. Show that if T is regular, then $\leq_T$ is a preordering, that is, is reflexive and transitive. A theory is said to be *algebraic* if $\leq_T$ is a partial ordering. Show that Th(A) is algebraic if and only if A is extensional.

9.3. Show that every preordering $\leq$ on a set Σ is $\leq_T$ for some regular theory T on Σ.

9.4. Let Σ be a set.
1. Identify the smallest regular theory T on Σ. Is it algebraic?
2. What is the largest regular theory on T? Is it algebraic?

9.5. Given a classification A show that each state description state$_A(a)$ is a partition of the set typ(A) of types of A. Prove that for all $a \in$ tok(A), a is a counterexample to the sequent $\langle \Gamma, \Delta \rangle$ if and only if $\langle \Gamma, \Delta \rangle \leq$ state$_A(a)$.

9.6. Let Γ and Δ be sets of types of a classification A. Show the following:
1. $\Gamma \vdash_A$ if and only if there is no token of A that is of all types in Γ;
2. $\vdash_A \Delta$ if and only if every token of A is of some type in Δ.

9.7. Given any set Σ, define a consequence relation on Σ by

$$\Gamma \vdash \Delta \quad \text{iff} \quad \begin{cases} \Gamma \cap \Delta \neq \emptyset, & \text{or} \\ \Gamma \text{ has more than one element}, & \text{or} \\ \Delta = \Sigma. \end{cases}$$

1. Show that $\vdash$ is the closure of the following set of sequents under Identity and Weakening:
   (a) $\langle \emptyset, \Sigma \rangle$.
   (b) $\langle \{\alpha, \beta\}, \emptyset \rangle$ for each pair α, β of distinct elements of Σ.
2. Show that $\vdash$ is closed under Global Cut.
3. Show that a classification A with typ(A) $= \Sigma$ is a complete state space if and only if for each $\Gamma, \Delta \subseteq \Sigma$, $\Gamma \vdash_A \Delta$ if and only if $\Gamma \vdash \Delta$.

9.2 Finite Cut and Global Cut

The results of this section are not needed in the remainder of the book. We include it in order to connect Global Cut with the more familiar version.

Definition 9.20. A theory $T = \langle \Sigma, \vdash \rangle$ is *compact* if for each constraint $\langle \Gamma, \Delta \rangle$ of T, there is a finite constraint of T such that $\langle \Gamma_0, \Delta_0 \rangle \leq \langle \Gamma, \Delta \rangle$; equivalently, if $\langle \Gamma, \Delta \rangle$ is a sequent and every finite sequent $\langle \Gamma_0, \Delta_0 \rangle \leq \langle \Gamma, \Delta \rangle$ is consistent, then so is $\langle \Gamma, \Delta \rangle$.

Example 9.21. Consider the theory of the truth classification of Example 9.3. The compactness theorem of first-order logic insures that this classification is compact. However, if we restricted the tokens to some class of structures not closed under first-order equivalence, like the class of well-orderings, then the resulting classification would not have a compact theory. Thus, in general, we cannot expect the theories we deal with to be compact.

The following result, whose proof is given as an answer to Exercise 9.8, shows that for compact theories, our notion of regularity agrees with the usual notion of structural rules of classical logic.

Proposition 9.22. *A compact theory T is regular if and only if it is closed under Identity, Weakening, and Finite Cut.*

The following gives an example of a theory that satisfies Identity, Weakening, and Finite Cut, but is far from the theory of any classification with a nonempty set of tokens.

Example 9.23. Let Σ be an infinite set. A subset Γ of Σ is *cofinite* if $\Sigma - \Gamma$ is finite. Define a consequence relation of Σ as follows:

$\Gamma \vdash \Delta$ if and only if either $\Gamma \cap \Delta \neq \emptyset$ or $\Gamma \cup \Delta$ is cofinite.

This relation $\vdash$ satisfies Identity, Weakening, and Finite Cut but is not regular. Indeed, the regular closure of this theory contains the absurd sequent $\vdash$ with both sides empty. (The proof of this claim is left as Exercise 9.9.) These observations show that in the case where a consequence relation is not compact, the rules of Identity, Weakening, and Finite Cut are far from sufficient for having a reasonable consequence relation.

Exercises

9.8. Prove Proposition 9.22.

9.9. Prove the claim made in Example 9.23.

9.10. The rule of *Infinite Cut* asserts that if $\Gamma, \Gamma' \vdash \Delta$ and $\Gamma \vdash \Delta, \alpha$ for each $\alpha \in \Gamma'$, then $\Gamma \vdash \Delta$. (There is also a version of Infinite Cut where the sides of the cut types are reversed. Everything said below about it applies to the other version without change.)

1. Show that every regular theory satisfies Infinite Cut.
2. Show that the consequence relation of Example 9.23 satisfies Infinite Cut.
3. Conclude that Identity, Weakening, and Infinite Cut do not entail Global Cut.

9.3 The Theory of a State Space

Each state space has a regular theory associated with it, because each state space is a classification, and a classification has a theory associated with it. From the informational point of view, however, the most natural classification to associate with a state space S is not the space itself, but its event classification $Evt(S)$.

By virtue of this classification, we can associate with S the regular theory $Th(Evt(S))$ generated by the event classification $Evt(S)$.

When state spaces are used in science, however, the tokens are not usually considered in any explicit manner. Thus, in order to be able to link up with the state space literature, we study a second regular theory on pow(typ(S)), one that is independent of the tokens of the state space. By way of introduction, we first note a property of all regular theories of the form $Th(Evt(S))$.

Proposition 9.24. *If S is a state space and $T = Th(Evt(S))$, then for all $\Gamma, \Delta \subseteq$ typ(Evt(S)), if $\bigcap \Gamma \subseteq \bigcup \Delta$ then $\Gamma \vdash_T \Delta$.*

Proof. Assume that $\bigcap \Gamma \subseteq \bigcup \Delta$. Let $s \in$ tok(S). We need to show that s satisfies $\langle \Gamma, \Delta \rangle$. So assume s is of every type $X \in \Gamma$. That is, $state_S(s) \in X$ for every such X. But then $state_S(s) \in \bigcap \Gamma$, so $state_S(s) \in \bigcup \Delta$. But then s is of type some $Y \in \Delta$, so s satisfies $\langle \Gamma, \Delta \rangle$. □

Notice this did not give us a biconditional. To get one, we must bring in the set Ω of "realized" states. A state is *realized* if it is the state of some token of the state space in question.

Proposition 9.25. *Given a state space S, let Ω be the set of realized states of S and let $T = Th(Evt(S))$. For each sequent $\langle \Gamma, \Delta \rangle$ of T, $\Gamma \vdash_T \Delta$ if and only if $(\bigcap \Gamma \cap \Omega) \subseteq \bigcup \Delta$.*

Proof. The proof of the direction from right to left is essentially the same as that of Proposition 9.24. To prove the other direction, assume $\Gamma \vdash_T \Delta$ and $\sigma \in (\bigcap \Gamma \cap \Omega)$ and let us prove that $\sigma \in \bigcup \Delta$. Because $\sigma \in \Omega$, σ is realized, so let s be of state σ. Because $\Gamma \vdash_T \Delta$, s satisfies the sequent $\langle \Gamma, \Delta \rangle$. But because s is of every type in Γ, it must be of some type in Δ so $\sigma \in \bigcup \Delta$. □

Corollary 9.26. *If S is a complete state space and $T = Th(Evt(S))$, then for all $\Gamma, \Delta \subseteq$ typ(T), $\bigcap \Gamma \subseteq \bigcup \Delta$ if and only if $\Gamma \vdash_T \Delta$. In particular, this holds for every ideal state space.*

Proof. To say that S is complete is to say that typ(S) is the set of realized states. □

When working with a state-space model of some phenomena, the default assumption is that the space is complete, that is, that every state is realized. After all, if one knew that some states were not realized, one would throw out