ESSLLI 2001, Helsinki, Finland

The Mathematics of Information

Lecture 4: Introduction to Channel Theory

Keith Devlin*

During the 1970s, Barwise and Seligman took the situation-theoretic approach to information and took away much of the technical machinery for handling situations, relations and types, to produce an abstract mathematical account of information flow.

Their motivating idea is that information flow is made possible by regularities in systems. Rather than develop machinery for analyzing those regularities, however, as situation theory does, instead they built a mathematical theory based on the mere existence of such regularities. The starting point is the notion of a classification.

Classifications

A *classification* is a structure $\boldsymbol{A} = \langle A, \Sigma_A, \models_A \rangle$, where A is a set of objects to be classified, called the *tokens* of $\boldsymbol{A}$, Σ_A is a set of objects used to classify the tokens, called the *types* of $\boldsymbol{A}$, and $\models_A$ is a binary relation between A and Σ_A which determines which tokens are classified by which types. We illustrate the classification relation as follows:

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

A familiar example to logicians is where the types are sentences of first-order logic and the tokens are mathematical structures, and $a \models \alpha$ is the relationship that the structure a is a model of the sentence α.

The first step is to develop machinery for discussing the "logic" of a system by means of which the system can support the flow of information.

Given a classification $\boldsymbol{A}$, a *sequent* is a pair (Γ, Δ) of sets of types of $\boldsymbol{A}$.

A token a of $\boldsymbol{A}$ is said to *satisfy* the sequent (Γ, Δ) if,

$$(\forall \alpha \in \Gamma)[a \models \alpha] \;\Rightarrow\; (\exists \alpha \in \Delta)[a \models \alpha]$$

*Center for the Study of Language and Information, Stanford University, Stanford, California 94305. devlin@csli.stanford.edu

We say that Γ *entails* Δ in $\boldsymbol{A}$, written $\Gamma \vdash_A \Delta$, if every token of $\boldsymbol{A}$ satisfies (Γ, Δ).

If $\Gamma \vdash_A \Delta$, then the pair (Γ, Δ) is said to be a *constraint* supported by the classification $\boldsymbol{A}$.

The set of all constraints supported by $\boldsymbol{A}$ is called the complete theory of $\boldsymbol{A}$, denoted by $\mathrm{Th}(\boldsymbol{A})$. The complete theory of $\boldsymbol{A}$ represents all the regularities supported by the system being modeled by $\boldsymbol{A}$.

Notice the following special cases of constraints:

- If α and β are both singletons, then $\alpha \vdash \beta$ is the claim that α logically entails β.
- The constraint $\vdash \alpha$, where the left-hand side is empty and α is a singleton, is the claim that α is necessarily true.
- The constraint $\alpha \vdash$, where the right-hand side is empty and α is a singleton, is the claim that no token is of type α.
- The constraint $\vdash \alpha, \beta$, where α, β is a doubleton, is the claim that every token is of (at least) one of the types α, β.
- The constraint $\alpha, \beta \vdash$, where the right-hand side is empty and the left-hand side is a doubleton, is the claim that the types α and β are mutually exclusive (i.e., no token is of type α and of type β).

Infomorphisms

In developing channel theory, Barwise and Seligman formulated three guiding principles concerning information flow:

- Information flow results from regularities in a distributed system.
- Information flow crucially involves both types and tokens.
- It is by virtue of regularities among connections that information about some components of a distributed system carries information about other components.

Let $\boldsymbol{A} = \langle A, \Sigma_A, \models_A \rangle$ and $\boldsymbol{C} = \langle C, \Sigma_C, \models_C \rangle$ be two classifications. An *infomorphism* between $\boldsymbol{A}$ and $\boldsymbol{C}$ is a pair $f = (f^\wedge, f^\vee)$ of functions that makes the following diagram commute:

$$\begin{array}{ccc} \Sigma_A & \xrightarrow{f^\wedge} & \Sigma_C \\ \Big| \models_A & & \Big| \models_C \\ A & \xleftarrow[f^\vee]{} & C \end{array}$$

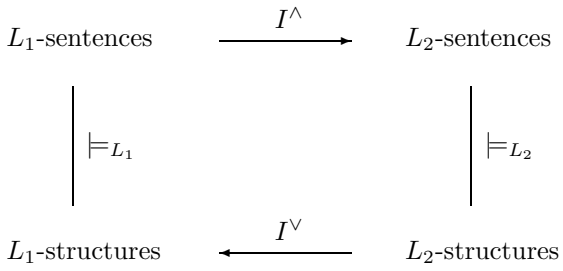
This means that

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

for all tokens c of $\boldsymbol{C}$ and all types α of $\boldsymbol{A}$. We refer to $f^\wedge$ as "f-up" and $f^\vee$ as "f-down". We take account of the fact that the functions $f^\wedge$ and $f^\vee$ act in opposite directions by writing

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

For example, suppose we have two mathematical theories M_1 and M_2, with corresponding first-order languages L_1 and L_2, respectively, and suppose further it is possible to interpret M_1 within M_2. (For example, M_1 could be arithmetic and M_2 set theory.) That is to say, with each structure m_2 of M_2 we can associate a unique structure $m_1 = I^\vee(m_2)$ of M_1, and each sentence α_1 of L_1 has a unique canonical translation $\alpha_2 = I^\wedge(\alpha_1)$ of L_2. (In the case where M_1 is arithmetic and M_2 set theory, $I^\vee(m_2)$ is the arithmetic in the model m_2 of set theory and $I^\wedge(\alpha_1)$ is the standard translation of α_1 into the language of set theory using the von Neumann definition of the natural numbers and the usual set-theoretic definitions of the operations of arithmetic.) Then $I = (I^\wedge, I^\vee)$ is an infomorphism and the following diagram commutes:



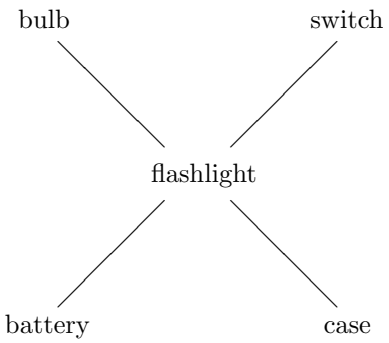
Based on the above example, given an infomorphism $f : \boldsymbol{A} \rightleftarrows \boldsymbol{C}$, we call $f^\wedge(\alpha)$ the *translation* of α and $f^\vee(c)$ the *particularization* of c.

Infomorphisms allow us to investigate the manner in which the parts of a system fit together to facilitate the flow of information. The key notion is that of an information channel.

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

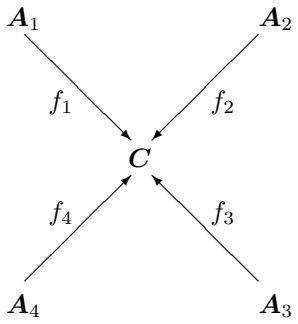
The intuition is that the $\boldsymbol{A}_i$ are individual components of the larger system $\boldsymbol{C}$, and it is by virtue of being parts of the system $\boldsymbol{C}$ that the constituents $\boldsymbol{A}_i$ can carry information about one another.

For example, consider a simple flashlight comprising a case, a bulb, a switch, and a battery. We can illustrate this system as follows:



The structure of the flashlight means that various components can carry information about others. For example, if the switch is of type ON and the battery is of type CHARGED then the bulb will be of type LIT.

In terms of an information channel, the flashlight can be represented like this:



Information flow

Based on the three basic principles they formulated (and other considerations), Barwise and Seligman proposed the following definition:

> Suppose $\boldsymbol{A}$ and $\boldsymbol{B}$ are constituent classifications in an information channel with core $\boldsymbol{C}$. A token a being of type α in $\boldsymbol{A}$ carries the information that a token b is of type β in $\boldsymbol{B}$ relative to the channel $\boldsymbol{C}$ if a and b are connected in $\boldsymbol{C}$ and the translation of α entails the translation of β in Th($\boldsymbol{C}$).

For example, in the case of the flashlight, suppose the classifications for the bulb, switch, battery, and case are everyday, common sense ones (bulbs have types LIT, UNLIT, and BROKEN, switches have types ON and OFF, etc.) and the classification of the flashlight is technical, involving principles of engineering, the laws of physics, etc. What does it mean to say that the switch of a particular flashlight being on carries the information that the bulb is lit?

Well, using the notation of the above diagram, the $\boldsymbol{A}_2$-type (i.e., the everyday switch-type) ON has a translation $f_2^\wedge$(ON) in $\boldsymbol{C}$. Likewise, the $\boldsymbol{A}_1$-type (i.e., the everyday bulb-type) LIT has a translation $f_1^\wedge$(LIT) in $\boldsymbol{C}$. The switch being on carries the information that the bulb is lit by virtue of the inference

$$f_2^\wedge(\text{ON}) \vdash_{\boldsymbol{C}} f_1^\wedge(\text{LIT})$$

Notice that the types in $\boldsymbol{C}$ provide the logical structure — the regularities — that gives rise to information flow, but information only flows in the context of a particular token c of $\boldsymbol{C}$ (i.e., a particular flashlight), for this is what provides the specific connections required to facilitate information flow. Specifically, if switch s is connected to bulb b by flashlight c, then s being on carries the information that b is lit.

We may further elucidate the way channels work by showing that the above definition of information flow satisfies Dretske's principle of veridicality: if $a \models_{\boldsymbol{A}} \alpha$ carries the information that $b : \beta$, then $b \models_{\boldsymbol{B}} \beta$. To see this, let c be the token in $\boldsymbol{C}$ that connects a and b. (The down-components of the relative infomorphisms map c to a and b.) If α' is the translation of α and β' that of β (i.e., the images of α and β under the up-components of the infomorphisms), then c must satisfy $\alpha' \vdash_{\boldsymbol{C}} \beta'$. Now, since c is the image of a under an infomorphism, $c \models_{\boldsymbol{C}} \alpha'$. Hence, applying the inference in $\boldsymbol{C}$, $c \models_{\boldsymbol{C}} \beta'$. Thus, applying the infomorphism that takes us from $\boldsymbol{C}$ to $\boldsymbol{B}$, $b \models_{\boldsymbol{B}} \beta$, as claimed.

To show that the Barwise-Seligman definition satisfies Dretske's Xerox Principle, we need a method for combining classifications.

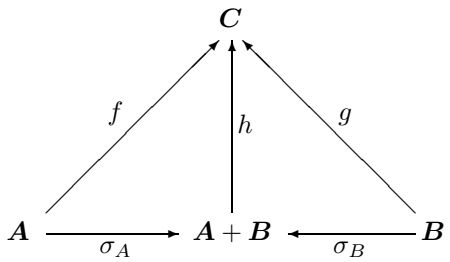
Given classifications $\boldsymbol{A}$ and $\boldsymbol{B}$, we define the *colimit* $\boldsymbol{A}+\boldsymbol{B}$ as follows. The tokens of $\boldsymbol{A}+\boldsymbol{B}$ consist of pairs (a, b) of tokens from each. The types of $\boldsymbol{A}+\boldsymbol{B}$ consists of of the types of both,

except that, if there are any types in common, we make two distinct (indexed) copies in order not to confuse them.

There are natural infomorphisms $\sigma_A : \boldsymbol{A} \rightleftarrows [\boldsymbol{A} + \boldsymbol{B}]$ and $\sigma_B : \boldsymbol{B} \rightleftarrows [\boldsymbol{A} + \boldsymbol{B}]$ defined thus:

1. $\sigma_A^\wedge(\alpha) = \alpha_A$ (the $\boldsymbol{A}$-copy of α), for each type α of $\boldsymbol{A}$.
2. $\sigma_B^\wedge(\beta) = \beta_B$, for each type β of $\boldsymbol{B}$.
3. for each token (a, b) of $\boldsymbol{A} + \boldsymbol{B}$, $\sigma_A^\vee((a, b)) = a$ and $\sigma_B^\vee((a, b)) = b$.

The name "colimit" comes from category theory. The classification $\boldsymbol{A} + \boldsymbol{B}$ has the property that, given any classification $\boldsymbol{C}$ and infomorphisms $f : \boldsymbol{A} \rightleftarrows \boldsymbol{C}$, $g : \boldsymbol{B} \rightleftarrows \boldsymbol{C}$, there is a unique infomorphism $h = f + g$ such that the following diagram commutes:



[Remember that an infomorphism consists of a pair of functions going in opposite directions. Commutativity in this case applies to both directions.]

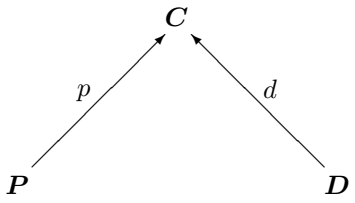
The definition of h is obvious: on tokens, $h^\vee(c) = (f^\vee(c), g^\vee(c))$; on types α_A, $h^\wedge(\alpha_A) = f^\wedge(\alpha)$; on types α_B, $h^\wedge(\alpha_B) = g^\wedge(\alpha)$.

To verify the Xerox Principle, now, if $a \models_{\boldsymbol{A}} \alpha$ carries the information that $b : \beta$ in $\boldsymbol{B}$, it does so by some channel with core $\boldsymbol{C}_1$. If $b \models_{\boldsymbol{B}} \beta$ carries the information that $d : \delta$ in $\boldsymbol{D}$, it does so by some channel with core $\boldsymbol{C}_2$. Let $\boldsymbol{C} = \boldsymbol{C}_1 + \boldsymbol{C}_2$. Then it is not hard to see that $a \models_{\boldsymbol{A}} \alpha$ carries the information that $d : \delta$ in $\boldsymbol{D}$ by way of the channel $\boldsymbol{C}$.

Reasoning at a distance

One obvious drawback with the approach so far is that is assumes we have complete information about $\mathrm{Th}(\boldsymbol{C})$, the theory of the channel core. This is fine for a theoretical analysis, but in practice this is generally not the case. Often, we use commonsense knowledge of the core in order to attribute information about one component to another. We now analyze such reasoning at a distance.

By way of illustration, consider a situation where an agent operating in a local environment uses local information in order to obtain information about and manipulate a distant environment. For example, an engineer in the control room of a nuclear power plant uses various monitors and gauges to obtain information about the reactor and uses switches and dials to control the reactor. In this case there are three classifications: the proximal classification $\boldsymbol{P}$, the distal classification $\boldsymbol{D}$, and the channel classification $\boldsymbol{C}$ that connects the two. The situation is illustrated below:



In using information in $\boldsymbol{P}$ to reason about $\boldsymbol{D}$, the operator makes (implicit) use of the infomorphisms $p : \boldsymbol{P} \rightleftarrows \boldsymbol{C}$ and $d : \boldsymbol{D} \rightleftarrows \boldsymbol{C}$. When we trace the information flow from a sensor in $\boldsymbol{D}$ to a gauge in $\boldsymbol{P}$, we use the infomorphism p from $\boldsymbol{P}$ to $\boldsymbol{C}$ followed by the infomorphism d backwards from $\boldsymbol{C}$ to $\boldsymbol{D}$.

We can develop the machinery we need to analyze this situation by considering a general situation of an infomorphism $f : \boldsymbol{A} \rightleftarrows \boldsymbol{B}$. Imagine someone who has to reason about tokens on one side using the natural theory of the other. We want to see how constraints (inferences) in one classification give rise to constraints in the other. That is, we need to formulate rules that tell us how to an inference in one classification corresponds to an inference in the other.

A good example to keep in mind is where $\boldsymbol{A}$ is Peano Arithmetic (PA) and $\boldsymbol{B}$ is set theory (say ZFC).

If Γ is a set of types of $\boldsymbol{A}$, we denote by Γ^f the set of translations of types in Γ.

If Γ is a set of types of $\boldsymbol{B}$, we denote by Γ^{-f} the set of types of $\boldsymbol{A}$ whose translations are in Γ.

The following two inference rules allow us to pass from one classification to another:

$$f\textbf{-Intro}: \quad \frac{\Gamma^{-f} \vdash_A \Delta^{-f}}{\Gamma \vdash_B \Delta}$$

$$f\textbf{-Elim}: \quad \frac{\Gamma^{f} \vdash_B \Delta^{f}}{\Gamma \vdash_A \Delta}$$

The first rule allows us to go from a sequent of $\boldsymbol{A}$ to a sequent of $\boldsymbol{B}$; the second rule allows us to go the other way round. In a moment we'll examine the validity of these rules, but first let's use the number theory and set theory example to get a sense of what they say.

In the case of number theory and set theory, f-**Intro** says that, if we take a valid sequent in PA, its translation into set theory is a valid sequent in ZFC. f-**Elim** says that if we take a sequent of set theory that happens to be a translation of a sequent in number theory, and if the sequent is valid in ZFC, then the original sequent is valid in PA.

Now what can we say about the validity of these two rues: what do they preserve?

f-**Intro** preserves validity. For if c were a counterexample to $\Gamma \vdash_B \Delta$, $f(c)$ would be a counterexample to $\Gamma^{-f} \vdash_A \Delta^{-f}$. This is obvious for the case of PA and ZFC.

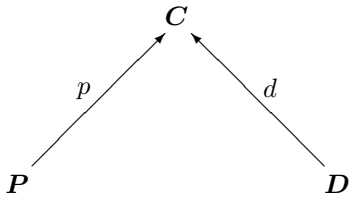
f-**Elim** does not preserve validity. There can be a valid constraint $\Gamma^f \vdash_B \Delta^f$ such that $\Gamma \vdash_A \Delta$ has a counterexample, although no counterexample can be of the form $f(c)$ for a token c of $\boldsymbol{B}$. For example, theorems of set theory in the language of number theory are reliable as long as we are only interested in models of number theory that are parts of models of set theory, but for other models the theorems are unreliable.

f-**Intro** does not preserve nonvalidity. For example, there are nontheorems of PA whose translations are theorems of ZFC; the consistency statement for PA is one such.

f-**Elim** does preserve nonvalidity. For instance, if a translation into set theory of a statement of number theory is false in ZFC, then the original statement must be false in PA.

Turning to systems now, the validity preserving nature of the f-**Intro** rule tells us that any constraint that holds for a component of a system translates to a constraint that holds for the system. Using f-**Elim**, however, we see that any constraint about the whole system gives 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.

In the case of the proximal–distal channel



we considered earlier, we can now examine what happens when we use the complete theory of the proximal classification $\boldsymbol{P}$ to reason about the distal classification $\boldsymbol{D}$.

The first step is to go from $\boldsymbol{P}$ to $\boldsymbol{C}$. Since p-**Intro** preserves validities, the translated theory we obtain on $\boldsymbol{C}$ is sound. But it may not be complete; there may be constraints of $\boldsymbol{C}$ that we miss.

Going from $\boldsymbol{C}$ to $\boldsymbol{D}$ now, using d-**Elim** means that we lose soundness (in addition to the completeness we lost in the first stage). 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 there are no guarantees about other distal tokens.

We shall track what is going on using the notion of a local logic. This generalizes the notion of the complete theory of a classification.

Local logics

A *local logic* $\mathcal{L} = \langle \boldsymbol{A}, \vdash_{\mathcal{L}}, N_{\mathcal{L}} \rangle$ consists of a classification $\boldsymbol{A}$, a set $\vdash_{\mathcal{L}}$ of sequents (satisfying certain structural rules) involving the types of $\boldsymbol{A}$, called the *constraints* of $\mathcal{L}$, and a subset $N_{\mathcal{L}} \subseteq \boldsymbol{A}$, called the *normal tokens* of $\mathcal{L}$, which satisfy all the constraints of $\vdash_{\mathcal{L}}$.

A local logic $\mathcal{L}$ is *sound* if every token is normal; it is *complete* if every sequent that holds of all normal tokens is in the consequence relation $\vdash_{\mathcal{L}}$.

A sound and complete local logic is essentially a classification. Using infomorphisms, however, we can move local logics around from one classification to another in a way that does not preserve soundness and completeness.

Given an infomorphism $f : \boldsymbol{A} \rightleftarrows \boldsymbol{B}$ and a logic $\mathcal{L}$ on one of these classifications, we obtain a natural logic on the other. If $\mathcal{L}$ is a logic on $\boldsymbol{A}$, then $f[\mathcal{L}]$ denotes the logic on $\boldsymbol{B}$ obtained from $\mathcal{L}$ by f-**Intro**. If $\mathcal{L}$ is a logic on $\boldsymbol{B}$, then $f^{-1}[\mathcal{L}]$ denotes the logic on $\boldsymbol{A}$ obtained from $\mathcal{L}$ by f-**Elim**.

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

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

where $\mathrm{Log}(\boldsymbol{P})$ 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 observed, it may be that $\mathrm{Log}_{\boldsymbol{C}}(\boldsymbol{D})$ is neither sound nor complete. But it is what is available in order to reason about $\boldsymbol{D}$ in $\boldsymbol{P}$.

It can be proved that every local logic on a classification $\boldsymbol{D}$ is of the form $\mathrm{Log}_{\boldsymbol{C}}(\boldsymbol{D})$ for some binary channel $\boldsymbol{C}$.

How information really flows

With the notion of a local logic available, let's now look again at Barwise and Seligman's definition of how information flows:

> Suppose $\boldsymbol{A}$ and $\boldsymbol{B}$ are constituent classifications in an information channel with core $\boldsymbol{C}$. A token a being of type α in $\boldsymbol{A}$ carries the information that a token b is of type β in $\boldsymbol{B}$ relative to the channel $\boldsymbol{C}$ if a and b are connected in $\boldsymbol{C}$ and the translation of α entails the translation of β in $\mathrm{Th}(\boldsymbol{C})$.

In many cases, $\boldsymbol{A}$ and $\boldsymbol{B}$ will consist of everyday folk theories (say of bulbs and switches in flashlights) whereas $\boldsymbol{C}$ will be a scientific/engineering theory — everything it takes to explain why flashlights work. Now, in reasoning about why the switch being moved to ON will carry the information that the light is LIT (or vice versa), a typical flashlight user (as opposed to a scientist or engineer) will use the folk theories. According to the above definition, however, the logic flow depends not on the folk theories themselves but on their translations into the scientific/engineering theory $\boldsymbol{C}$. Although this is arguably correct at a theoretical level — it is after all the science and the engineering that tells us exactly how flashlights work — this does not really capture the reasoning of the user. This we can do using local logics.

First, we note that the folk theory of how flashlights work is an amalgam of the folk theories of switches, bulbs, batteries, cases, and whatever. We represent this in our theory by using a natural extension of the colimit construction. Given any channel $\boldsymbol{C} = \{f_i : \boldsymbol{A}_i \stackrel{\rightarrow}{\leftarrow} \boldsymbol{C}\}_{i\in I}$, we can represent it by a single infomorphism $f : \boldsymbol{A} \stackrel{\rightarrow}{\leftarrow} C$ by taking the sum $\boldsymbol{A} = \sum_{i\in I} \boldsymbol{A}_i$ and the sum $f = \sum_{i\in I} f_i$ of the f_i. Given any logic $\mathcal{L}$ of the core, we can use f-**Elim** to obtain a local logic $f^{-1}[\mathcal{L}]$ on $\boldsymbol{A}$. It is this logic, with its constraints and normal tokens, that captures the information flow in the channel from the flashlight user's perspective. Or, as Barwise and Seligman put it [p.41], the local logic $f^{-1}[\mathcal{L}]$ is the "what" of information flow, the channel is the "why."

Let's look at the flashlight example in a bit more detail. We'll restrict our attention to the switch and the bulb for simplicity. Let $f : \boldsymbol{B} \stackrel{\rightarrow}{\leftarrow} \boldsymbol{F}$ represent the part-whole relation between bulbs ($\boldsymbol{B}$) classified in commonsense ways and flashlights ($\boldsymbol{F}$) classified scientifically. Likewise, let $g : \boldsymbol{S} \stackrel{\rightarrow}{\leftarrow} \boldsymbol{F}$ represent the part-whole relation between switches ($\boldsymbol{S}$) classified in commonsense ways and flashlights ($\boldsymbol{F}$) classified scientifically.

Putting these two infomorphisms together we get an infomorphism $h = f + g$ from $\boldsymbol{B} + \boldsymbol{S}$ to $\boldsymbol{F}$. Given a flashlight token x, $h(x) = (f(x), g(x))$, where $f(x)$ and $g(x)$ are the bulb and switch of x. Given a type Γ of $\boldsymbol{F}$, $h(\Gamma)$ is the disjoint union of $f(\Gamma)$ and $g(\Gamma)$.

Suppose now that $\boldsymbol{B}$ supports the constraint

$$\textsc{lit} \vdash_{\boldsymbol{B}} \textsc{live}$$

It is easily seen that this is a constraint of $\boldsymbol{B} + \boldsymbol{S}$. Then, whatever the classification $\boldsymbol{F}$ of flashlights is, and whatever h does to these types, we have

$$h(\textsc{lit}) \vdash_{\boldsymbol{F}} h(\textsc{live})$$

This is because h-**Intro** preserves validity.

Now let's go the other way round. Suppose the classification $\boldsymbol{F}$ supports the constraint

$$\textsc{illum} \vdash_{\boldsymbol{F}} \textsc{elec}$$

where ILLUM is the technical property of an electrical component emitting photons and ELEC is the technical property of a component carrying electric current. Thus $\textsc{illum} = h(\textsc{lit})$ and $\textsc{elec} = h(\textsc{on})$ (let us assume). Applying h-**Elim**, we get

$$\textsc{lit} \vdash_{\boldsymbol{B}+\boldsymbol{S}} \textsc{on}$$

However, this sequent is not valid. There are pairs (b, s) of switches and bulbs such that s is on but b is not lit. Indeed, there are many such pairs. The above inference only holds (qua an inference) for pairs that are tokens of the same flashlight. These pairs are the normal tokens of the logic obtained by h-**Elim**. This exemplifies our earlier observation that, in general, h-**Elim** does not preserve validity.

Handling exceptions

In examining the flashlight example, we assumed that all components were in working order. Thus, we did not question the validity of the constraint "If the switch is on, then the bulb is lit." But, of course, in real life this is not necessarily true. For instance, this constraint fails if the battery is dead. This is an example where the weakening rule

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

fails. (It is generally valid for mathematical reasoning.) The validity of the constraint

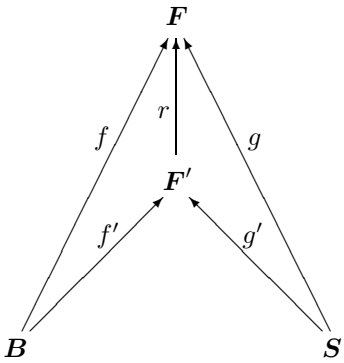
The switch being on entails the bulb is lit

does not imply the validity of the constraint

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

The channel-theoretic explanation for this phenomenon is that the introduction of an exception such as the battery being dead changes the channel to one where the relevant classifications have more types. Within a given channel, the Rule of Weakening does in fact hold.

For example, suppose we analyze the flashlight in terms of channels $\mathcal{F}$ and $\mathcal{F}'$ having core classifications $\boldsymbol{F}$ and $\boldsymbol{F}'$, respectively. Suppose that $\mathcal{F}'$ and $\mathcal{F}$ are identical apart from the fact that $\mathcal{F}'$ contains tokens where there are dead batteries but $\mathcal{F}$ does not. Then $\mathcal{F}'$ is a *refinement* of $\mathcal{F}$, which means that there is an infomorphism $r : \boldsymbol{F}' \rightleftarrows \boldsymbol{F}$ such that the following diagram commutes



(Take r to be the identity on both types of $\mathcal{F}'$ and tokens of $\mathcal{F}$.)

Using r-**Intro**, any constraint of $\boldsymbol{F}'$ will yield a constraint of $\boldsymbol{F}$; i.e., any sequent that holds in $\mathcal{F}'$ will hold in $\mathcal{F}$. However, since r-**Elim** does not preserve validity, there may be constraint of $\mathcal{F}$ that are not valid sequents in $\mathcal{F}'$. This will in fact be the case if there is a flashlight x that has a dead battery. Relative to the channel $\mathcal{F}$, a flashlight switch being closed does carry the information that the bulb is lit; relative to the channel $\mathcal{F}'$, however, this is not the case. In symbols

$$h(\textsc{on}) \;\vdash_{\boldsymbol{F}}\; h(\textsc{lit})$$

but

$$h'(\textsc{on}) \;\not\vdash_{\boldsymbol{F}'}\; h'(\textsc{lit})$$

(where $h' = f' + g'$).

What channel theory does not do

Channel theory does an excellent job in achieving its aim: to provide a mathematical model of information flow that captures the way agents reason about the world using partial information. However, it was not developed as a tool to be used directly in real world reasoning. In that respect, it is very much like Dretske's theory that preceded it. In contrast, Shannon's theory and situation theory were both developed to be used — by communications engineers in the case of Shannon's theory and by social and computing scientists in the case of situation theory. By eliminating the machinery for handling types and constraints, channel theory was able to develop as an elegant mathematical theory. But it is, of course, precisely the types and constraints apparatus that are required for analyzing actual instances of information flow in the real world. Thus, situation theory and channel theory provide an excellent complementary pair of linked ways to approach information flow.

If this were a mathematics meeting, in my final lecture I would go into the mathematics of channel theory, proving various theorems about the theory. For this audience, however, I shall devote the final lecture to some applications of situation theory.

Reference

Barwise, Jon and Seligman, Jerry, *Information Flow: The Logic of Distributed Systems*, Cambridge University Press (1997).