This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIT.2020.2987806, IEEE Transactions on Information Theory

Information Flow in Computational Systems

Praveen Venkatesh*, Sanghamitra Dutta† and Pulkit Grover‡
Electrical & Computer Engineering, and the Center for the Neural Basis of Cognition
Carnegie Mellon University
*vpraveen@cmu.edu †sanghamd@andrew.cmu.edu ‡pulkit@cmu.edu

***Abstract*—We develop a theoretical framework for defining and identifying flows of information in computational systems. Here, a computational system is assumed to be a directed graph, with "clocked" nodes that send transmissions to each other along the edges of the graph at discrete points in time. We are interested in a definition that captures the dynamic flow of information about a specific *message*, and which guarantees an unbroken "information path" between appropriately defined inputs and outputs in the directed graph. Prior measures, including those based on Granger Causality and Directed Information, fail to provide clear assumptions and guarantees about when they correctly reflect information flow about a message. We take a systematic approach—iterating through candidate definitions and counterexamples—to arrive at a definition for information flow that is based on conditional mutual information, and which satisfies desirable properties, including the existence of information paths. Finally, we describe how information flow might be detected in a noiseless setting, and provide an algorithm to identify information paths on the time-unrolled graph of a computational system.**

I. Introduction

A. Motivation

Neuroscientists often seek an understanding of how information flows in the brain while it performs a particular task [2]–[6]. As a concrete example, consider the experiment performed by Almeida et al. [2], where they examine how images of common handheld tools are processed in the brain. In simple terms, the question they investigate is this: when attempting to identify a handheld tool, does one make use of knowledge of how to manipulate it? Two hypotheses present themselves: (i) the answer to the above question is *yes*, so we should expect that information about a tool's identity *first* flows from visual cortex to motor cortex (the area responsible for processing manipulation), *before* synthesis of visual and motor information occurs at the area of the brain responsible for object recognition; (ii) alternatively the answer to the aforementioned question is *no*, so we should expect that the information about tools' identities *first* flows from visual cortex to the area responsible for object recognition, *after* which this information arrives at motor cortex. Thus, distinguishing between these hypotheses is equivalent to *determining the path* along which information about a tool's identity flows in the brain. What methods can neuroscientists use to gain such an understanding? What formal theory underlies such an analysis? How does one mathematically define colloquially-used terms such as "information flow"? These are the fundamental questions we try to answer in this paper.

As another example, consider the work of Hong et al. [3], who show that mice can detect the presence of an object with their whiskers, even without their "barrel cortex", the primary sensory area of the brain responsible for this task. It is hypothesized that information about the object's presence passes from neurons at the whiskers through an alternative pathway involving deeper brain regions that add redundancy to the system. How is sensory information about the presence of external objects encoded between the cortex and these deeper regions? How much of the information flow is expressed uniquely in each area, and how much of it is redundant? And once again, how do we systematically discuss the measurement of information flow along each of these pathways?

Information flow is a concept that appears in several contexts, across fields ranging from communication systems [7], [8], control theory [9], [10] and neuroscience [2]–[6] to security [11], algorithmic transparency [12], [13], and deep learning [14]–[16]. While our primary motivation comes from neuroscience, the theory that we develop is broadly applicable to any system which can be modeled in the form of a directed graph, with nodes that communicate functions of their inputs to other nodes, and where transmissions are observable. For example, several kinds of social networks readily fit this bill, and one might wish to analyze the spread of (mis)information or infectious disease in such networks [17, Ch. 16]. Our framework is also general enough to analyze information flow in various kinds of Artificial Neural Networks: this could be useful for identifying specific paths that carry information distinguishing two or more classes, or for intelligently pruning an Artificial Neural Network post-training [18], [19].

In the field of neuroscience, studying the paths along which information flows could be essential for understanding, diagnosing and treating brain diseases [20]–[23]. For example, understanding information flow pathways is essential when considering principled approaches for intervening to affect the output of a computational system, as is done in Responsive Neurostimulation in Epilepsy [24] or Deep Brain Stimulation in Parkinson's Disease [25], or for complementing dysfunctional aspects of the nervous system, such as in cochlear [26] and retinal implants [27].

B. Our Goal and Approach

Our overarching goal in this paper is develop a formal theory for understanding information flow in neuroscientific

A short version of this paper has appeared in the 2019 IEEE International Symposium on Information Theory [1].

experiments. Based on the examples highlighted in the previous section, we can summarize the key aspects of what we mean by "information flow". We want to capture:

1) information flow between *distinct nodes*, representing different brain regions at some scale;
2) information about a *specific message*, i.e., the stimulus;
3) specific *paths* along which information flows;
4) *dynamics*, i.e., there may be information flow at one time instant, but not at a later instant; and
5) *feedback*, i.e., information about a message may flow back and forth between two areas.

In what follows, we expand upon some of these points, and outline our methodology for designing a computational model and providing a definition of information flow that addresses these issues.

In order to properly scope our task, we choose to restrict our attention to "*event related experimental paradigms*" [28]. These are a set of standard neuroscientific experimental protocols where a timed *stimulus* is presented to an animal subject or human participant over multiple trials, while their brain signals are being recorded. Restricting ourselves to such experiments allows us to decide precisely what *kind* of information flow we are interested in, since in general, the phrase "information flow" can refer to more than one notion in neuroscience. We identify two dominant interpretations of "information flow": (i) the first refers to information about a *specific* quantity or variable that is of interest to the experimentalist, which in this paper we refer to as the *"message"*; (ii) the second refers to information in the *abstract*, and is usually used to describe the fact that one area of the brain "drives" or "influences" another area through the transmission of some information: in this interpretation, one is not interested in *what* is being communicated, only that the communication is *occurring*. In this paper, we focus only on the first interpretation of the phrase, where we are interested in information about a *specific message*. This is particularly common in event-driven paradigms, where the neuroscientist investigates how the brain responds to a carefully chosen set of stimuli, and examines the paths along which information contained in these stimuli[1] flows through the brain.

We approach our goal of providing a theoretical framework for information flow by formally defining a computational system model. This model is based on a graph consisting of nodes, representing distinct computational areas of the brain, and edges, representing the connections between them. The nodes of this graph can potentially represent the brain at any scale: single neurons, groups of neurons, or even whole brain regions, depending on the measurement modality and the kind of experiment being performed. These nodes compute stochastic functions of transmissions received from their incoming edges, and send the results of these computations on their outgoing edges. The idea of the computational graph is inspired from Thompson's work on VLSI complexity theory [29], while the model for stochastic computation is derived from Structural Causal Models in the field of Causality [30], [31].

[1]or alternatively, information contained in the *response*

Next, we find a formal definition for information flow that satisfies the requirements listed above. In order to attain a *dynamic* picture of information flow on the edges of the computational graph, and to deal with *feedback* in the flow of information, we use the idea of time-unrolling the computational graph (taking inspiration from Network Information Theory [7]). Principal among our requirements is that we must be able to *track, using an unbroken path, how information about the message flows through the system*. Imposing this requirement as a desired property, we iterate through a series of candidate definitions and counterexamples, finally arriving at a definition based on conditional mutual information, which satisfies the aforementioned requirements.

Given that we are interested in *information about a message*, we rely on information-theoretic measures to define information flow. Furthermore, we restrict ourselves to "*observational*" measures, which can be computed from a sample of all random variables described in the model. We deliberately eschew *interventional* and *counterfactual* measures: the former require the capability to intervene on the system and change the distributions of the random variables involved; meanwhile the latter are usually theoretical, and can only be applied in situations where one can ask what *might have occurred* if a specific variable had been different on a particular trial (while keeping the realizations of all unobserved private sources of randomness fixed).

The approach of building a rigorous theoretical framework that we have adopted in this paper is inspired by two works from biologists titled "Can a biologist fix a radio?" [32] and "Could a neuroscientist understand a microprocessor?" [33]. Both these works point to the lack of formal methods, i.e., systematic theory, that could help biologists understand the limitations of their tools and test their assumptions. It is our belief that information theory can help provide the formal methods that are sought in biology, and make an impact in fields such as neuroscience and neuroengineering [34]–[36]. In particular, information theory can play an important role in advancing how we understand large computational systems through external measurements and interventions. While developing an understanding of information flow in such systems may not be sufficient for providing a complete description of the nature of computation itself, we believe that it forms an integral component. Providing a formal theoretical framework for information flow is but a small part of several broader theoretical questions that are yet to be properly posed: questions such as how one might formalize "reverse engineering" the brain, or formalize the notion of "understanding computation".

C. Related Work

Prior work on statistically inferring flows of information in the brain appears under the umbrella of "functional" or "effective connectivity" [37]–[39]. These efforts have largely relied on measures of statistical[2] causal influence such as Granger Causality [40], [41], Massey's Directed Information [42]–[45], Transfer Entropy [46] and Partial Directed

[2]We borrow the use of the term "statistical" from Pearl [30, Sec. 1.5], who contrasts and differentiates "statistical" concepts from (strictly) "causal" ones.

Coherence [47]. Despite widespread use, these measures have frequently been a subject of debate and disagreement within the neuroscientific community [48]–[54]. In part, these disagreements stem from the widely-acknowledged fact that under non-ideal measurement conditions (e.g. in the presence of hidden variables [30, p. 54], asymmetric noise [55], [56], or limited sampling [57]), estimation of these quantities may be erroneous. While these non-idealities may eventually be overcome through improvements in technology, we believe that more fundamental issues still remain. For instance, one basic question that has remained unanswered is: when can statistical causal influence be interpreted as information flow about a message? In previous work, we demonstrated that even under *ideal* measurement conditions, the direction of greater Granger causal influence can be opposite to the direction in which the message is being communicated in certain kinds of feedback communication networks [58]. This example points to a more general issue with the use of statistical causal influence measures: there is no direct way to interpret what the influence is *"about"*. While it is understood in certain settings that "information flow" refers to information contained in a particular set of *"stimuli"* (as mentioned in the previous section), the aforementioned measures do not incorporate the effect of the stimulus.

The existence of such fundamental issues can be traced back to the fact that there is *no underlying model* that links information flow (of some message of interest) with the signals that are actually *measured*. This leads to a lack of separation between the problems of *defining* information flow and of *estimating* it, while also making it hard to test assumptions and draw the right interpretations from experimental analyses. We believe that, just as Shannon provided a theoretical foundation for information transmission [59], a solid theoretical treatment of information flow is needed. Such a treatment would begin with a model of the underlying system, give a definition for information flow and describe its properties, and finally end with a suitable estimator. Adopting Shannon's approach of defining entropy by stating a set of properties that such a measure must satisfy, we attempt to define information flow by putting forward an intuitive property that we believe is desirable for such a quantity. It is our hope that, by providing a theoretical foundation that separates definition and estimation, along with a concrete model and explicitly-stated assumptions, we can avoid many of the pitfalls encountered by previous approaches to understanding information flow in the brain.

It is useful at this point to mention the key differences between our measure of information flow, and measures based on Granger Causality and its generalizations:

1) Our measure depends on a message, M, that is related to the stimulus or the response in a neuroscientific task, whereas tools based on Granger Causality do not.
2) Since Granger Causality-based tools use time series modeling to compute an estimate of information flow, they are unable to provide a dynamic, evolving picture of information flow between different areas over time (although we must acknowledge that there have been recent efforts towards bridging this gap [60]).
3) Since we start with a *computational framework*, our model provides a direct way to connect information flow with the underlying computation. On the other hand, Granger Causality-based tools start with a probabilistic graphical model of the observed nodes, and do not tie the analysis to computation in any way.

While our proposed definition of information flow will also suffer from performance degradation under non-ideal measurement conditions, we believe that it overcomes the fundamental difficulty faced by Granger Causality-based tools: when measurements are ideal, our definition provides a clear and consistent way to interpret information flow about a message, as we illustrate through several examples in Section VI.

Another line of work that appears within the functional and effective connectivity literature is Dynamic Causal Modeling (DCM) [38], [61]. This methodology is, in spirit, much more closely aligned with what we propose here. However, our framework differs from DCM in a few important ways: (i) our underlying framework and model is based on Structural Causal Models rather than dynamical systems, and (ii) we seek to formalize the notion of information flow, not just of effective connectivity. However, the style of thinking, which involves starting from theoretical models and incorporating the stimulus and experimental design, is common to both DCM and our approach.

D. Outline of the Paper

In this paper, we start by giving a mathematical description of a generic computational system, about which inferences are being drawn (Section II). We then formally define what it means for information about some message to flow on a single edge or on a set of edges in the computational system (Section III). This is done by proposing an intuitive property that we would like such flows to satisfy, along with some candidate definitions, and then examining which candidates satisfy the property. The intuitive property we desire is: *information flow about a message may not completely disappear from the system at a certain time, only to spontaneously reappear at a later point* (formalized in Property 1). It emerges that simple and intuitive definitions actually fail to satisfy this basic property, and so a more sophisticated definition is needed. We then show how our definition for information flow about the message satisfies several desirable properties, including guarantees for the existence of "information paths" between appropriately defined input and output nodes (Section IV). We also show how our definition has a very non-intuitive property—information about a message may flow *out* of a node despite not flowing *into* it—and justify why this might be reasonable for an observational definition. After that, we suggest how one might detect which edges of the computational system have information flow, and provide an "information path algorithm", which identifies the aforementioned information paths (Section V). We also introduce and discuss the concepts of derived information, redundant transmissions and hidden nodes, which allow one to obtain a more fine-grained understanding of information structure in the computational system. To show that our definition of information flow agrees with intuition, we give several canonical examples of

computational systems and depict the information flow in each case (Section VI). Finally, we conclude with discussions on connections with neuroscience, issues related to the difficulty of estimating information flow (along with possible remedies), comparisons with the existing statistical causal influence literature, connections with fields such as probabilistic graphical models and causality, and a discussion on information volume (Section VII).

II. The Computational System

Our goal is to develop a rigorous framework for understanding how information about a message flows in a computational system. To do this, we first need to define the terms "computational system", "message", "information about a message" and "flow". In this section, we start with the first two terms, defining the model of the computational system that is used throughout this paper, and explicitly defining the message.

Our model is based on prior art in the information theory literature [7], [29], and consists of nodes communicating to each other at discrete points in time on a directed graph. At every time instant, each node receives transmissions on its incoming edges and computes a function of these transmissions to send out on its outgoing edges. This function can be random and time-dependent, and can be different for every outgoing edge. We will be interested in the flow of a particular random variable called the "message", which will be defined shortly. Since the directed graph forming the computational system may have cycles, the message may flow along a cyclic path. To deal with this possibility while capturing the fact that nodes must be causal,[3] we define a "time-unrolled" graph (in a manner similar to Ahlswede et al.[4] [7]), which describes how nodes communicate to each other over time. We define a random variable model for the nodes' transmissions, and demonstrate how each node computes these variables. We also formally define the input nodes of the computational system, through their relationship with the message.

Definition 1 (Complete directed graph): A *complete directed graph* $\mathcal{G}^* = (\mathcal{V}^*, \mathcal{E}^*)$ is described by a set of nodes and the set of all edges between those nodes (including self-edges). We denote the set of nodes by their indices, $\mathcal{V}^* = \{1, 2, \ldots, N\}$, where N is a positive integer denoting the number of nodes in the graph. The set of edges in the graph is the set of all ordered pairs of nodes, $\mathcal{E}^* = \mathcal{V}^* \times \mathcal{V}^*$.

Note that (i) edges are directed, so the edge $(A, B) \in \mathcal{E}^*$ describes an edge *from* node A *to* node B; and (ii) nodes have self-edges. For every $A \in \mathcal{V}^*$, there is an edge (A, A) in $\mathcal{E}^*$.

Moving forward, nodes shall be thought of as performing computations and possessing local memories. We shall interpret the transmission of a node to itself as the variable it stores within its memory.[5]

Definition 2 (Time-unrolled graph): In order to allow nodes to have different transmissions at every time instant, we must provide for the progression of time. Let $\mathcal{T} = \{0, 1, \ldots, T\}$ be a set of time indices, where T is a positive integer representing the maximum time index. Then, a *time-unrolled graph* $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is constructed by indexing a complete directed graph $\mathcal{G}^*$ using the time indices $\mathcal{T}$ as follows:

1) The nodes $\mathcal{V}$ consist of all nodes $\mathcal{V}^*$ in $\mathcal{G}^*$, subscripted by time indices in $\mathcal{T}$,
$$\mathcal{V} = \{A_t : A \in \mathcal{V}^*, t \in \mathcal{T}\};$$
2) The edges $\mathcal{E}$ connect nodes of *successive* times in $\mathcal{V}$, so they can be written in terms of the edges in $\mathcal{E}^*$ as
$$\mathcal{E} = \big\{(A_t, B_{t+1}) : (A, B) \in \mathcal{E}^*, t \in \mathcal{T} \setminus \{T\}\big\}.$$

For brevity, we denote the set of all nodes at time t by $\mathcal{V}_t$, and the set of all (outgoing) edges at time t by $\mathcal{E}_t$. So, for example, we will have $A_1 \in \mathcal{V}_1$ and $(A_1, B_2) \in \mathcal{E}_1$. All of the notation in this section can be visualized in Figure 1 and is summarized in Table I.

Once again, note that (i) edges at time t connect nodes at time t to nodes at time $t+1$; and (ii) since the original graph $\mathcal{G}^*$ had self-edges, there will always be an edge (A_t, A_{t+1}) in $\mathcal{E}_t$ for every node $A_t \in \mathcal{V}_t$.

Also note, we have only presented the complete directed graph in Definition 1 in order to explicitly define the process of time-unrolling. We do not expect the time-unrolled graph to be "rolled back" into a complete directed graph at the end of an information flow analysis. Since we seek a time-evolving picture of information flow between different computational nodes, we will directly view and interpret information flow on the time-unrolled graph. This is illustrated later, through several examples, in Section VI.

Definition 3 (Computational System): A *computational system* $\mathcal{C} = (\mathcal{G}, X, W, f)$ is a time-unrolled graph $\mathcal{G}$ that has *transmissions on its edges* which are constrained by *computations at its nodes*. The *input* to the computational system includes a *message*,[6] M. We now elaborate upon these terms:

3a) Transmissions on Edges

We begin by defining a function which maps every edge of $\mathcal{G}$ to a random variable. Let $\mathfrak{X}$ be the set of all random variables in some probability space.[7] Then, let $X : \mathcal{E} \to \mathfrak{X}$ be a function that describes what random variable is being transmitted on a given edge, i.e., $X(E)$ is the random variable corresponding to the transmission on the edge E.

[3]Causal in the "Signals and Systems" sense of the word, where a node cannot make use of future transmissions [62].

[4]Although the work of Ahlswede et al. (2000) is titled "Network *Information Flow*", it actually addresses a different problem: one of the achievable rate region of a broadcast network and the optimal coding strategy that achieves this rate. In contrast to their work, which concentrates on characterizing and achieving the optimal rate, our focus is on understanding how information about a known message flows in an existing computational system.

[5]Instances of directed graphs that are *not complete* and of nodes possessing *no memory* are merely special cases of our model, where the respective edges' transmissions can simply be set to zero.

[6]The message is the random variable *whose* "information flow" we will seek to identify.

[7]We assume that all probability distributions are such that the mutual information and conditional mutual information between any sets of random variables is well-defined [63, Sec. 2.6].

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIT.2020.2987806, IEEE Transactions on Information Theory

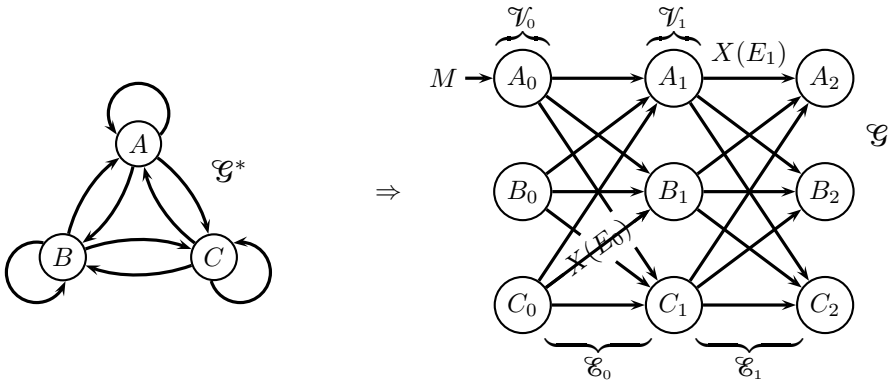


Figure 1: A diagram showing an example of a how a complete directed graph is unrolled to create a time-unrolled graph. On the left, we show a complete directed graph $\mathscr{G}^*$ that has three nodes, $\mathscr{V}^* = \{A, B, C\}$. These nodes are fully connected to each other via edges $\mathscr{E}^*$, including self-edges.
On the right, we show how $\mathscr{G}^*$ has been unrolled using time indices $\mathscr{T} = \{0, 1, 2\}$ to obtain a time-unrolled graph $\mathscr{G}$. The set of all nodes at time $t = 0$ is $\mathscr{V}_0$ and the set of all (outgoing) edges at time $t = 0$ is denoted $\mathscr{E}_0$. As an example, we have shown an arbitrary edge $E_0 \in \mathscr{E}_0$ (here, $E_0 = (C_0, B_1)$) and the transmission on that edge, $X(E_0)$. As another example, we show a "self-edge" in the time-unrolled graph, $E_1 \in \mathscr{E}_1$, which in this case is $E_1 = (A_1, A_2)$. Also depicted is the transmission $X(E_1)$ on this self-edge, which is interpreted as the contents of the memory of node A from $t = 1$ to $t = 2$. The message M arrives at the input node A_0, but could in general be available at more than one node at $t = 0$.
In subsequent illustrations, we do not depict all edges at every time step, even though they are present. This is done only for the sake of clarity.

For convenience, we define X applied to a *set of edges* as the set of random variables produced by applying X to each of those edges individually, i.e., for any set $\mathscr{E}' \subseteq \mathscr{E}$,

$$X(\mathscr{E}') = \{X(E) : E \in \mathscr{E}'\}. \tag{1}$$

We extend the use of this notation to other functions of nodes and edges that we define, going forward.

3b) Computation at a Node

Let $A_t \in \mathscr{V}_t$ be a node in the time-unrolled graph $\mathscr{G}$, at some time $t \geq 1$ (recall that $t \in \{0, 1, \ldots, T\}$). Let $\mathscr{P}(A_t)$ be the set of edges entering A_t, and $\mathbb{Q}(A_t)$ be the set of edges leaving A_t. Further, let us suppose that A_t is able to intrinsically generate the random variable[8] $W(A_t)$ at time t, where $W(A_t) \perp\!\!\!\perp W(\mathscr{V} \setminus \{A_t\}) \ \forall \ A_t \in \mathscr{V}$, $W(\mathscr{V}_t) \perp\!\!\!\perp \{M\} \cup \{X(\mathscr{E}_{t'}) : t' \in \mathscr{T}, t' < t\}$ and the symbol "$\perp\!\!\!\perp$" stands for independence between random variables. Then, the *computation* performed by the node A_t (for $t \geq 1$) is a deterministic function[9] f_{A_t} that satisfies

$$f_{A_t}\big(X(\mathscr{P}(A_t)), W(A_t)\big) = X(\mathbb{Q}(A_t)). \tag{2}$$

Here, $X(\mathscr{E}_{t-1})$, $W(\mathscr{V} \setminus \{A_t\})$, $W(\mathscr{V}_t)$, $X(\mathscr{P}(A_t))$ and $X(\mathbb{Q}(A_t))$ all make use of the notation described in (1).

Note that the definition above does not apply when $t = 0$; this is a special case which is discussed below. Also, for convenience, where $\mathscr{A}$ is an arbitrary set of nodes, we will use $f_{\mathscr{A}}$ to denote the "joint function" mapping the incoming transmissions of all nodes in $\mathscr{A}$ (along with their intrinsic random variables $W(\mathscr{A})$) to their respective outgoing transmissions.

[8] $X(E_t)$ and $W(A_t)$ may also be random *vectors* instead of random variables, i.e., an edge may *transmit a vector*. This does not affect the theoretical development presented here; all of our proofs remain unchanged.

[9] This kind of model is not new. For instance, in the causality literature, it is known by a few different names: Pearl refers to it as a "Structural Equation Model" [30, Sec. 1.4.1], while Peters et al. refer to it as a "Structural Causal Model" [31]. We prefer the latter terminology, which makes explicit the connection to causality.

3c) The Message and the Input Nodes

The *message* is a random variable M, which is of interest to the experimentalist observing the computational system, and for which we shall define information flow. For now, we assume that we are interested in a single message.[10] We also assume that the message enters the computational system only at time $t = 0$, and at no later time instant.

We formally define the *input nodes* of the system as those nodes of $\mathscr{G}$, at time $t = 0$, whose transmissions statistically depend on the message M:

$$\mathscr{V}_{\text{ip}} := \{A_0 \in \mathscr{V}_0 : I\big(M; X(\mathbb{Q}(A_0))\big) > 0\}, \tag{3}$$

where $\mathbb{Q}(A_0)$ represents the set of edges leaving the node A_0.

To remain consistent with Definition 3b, we define the computation performed by an input node $A_0 \in \mathscr{V}_{\text{ip}}$ as a function f_{A_0} that satisfies

$$f_{A_0}\big(M, W(A_0)\big) = X(\mathbb{Q}(A_0)), \tag{4}$$

and the computation performed by a non-input node at time $t = 0$, $A_0 \in \mathscr{V}_0 \setminus \mathscr{V}_{\text{ip}}$, as a function f_{A_0} that satisfies

$$f_{A_0}\big(W(A_0)\big) = X(\mathbb{Q}(A_0)). \tag{5}$$

As before, $W(A_0) \perp\!\!\!\perp W(\mathscr{V}_0 \setminus \{A_0\})$ for all $A_0 \in \mathscr{V}_0$ and $W(\mathscr{V}_0) \perp\!\!\!\perp M$.

Remarks:

1) Informally speaking, Definition 3 is designed to allow each node to generate a randomized function of its incoming transmissions for each of its outgoing transmissions.
2) The randomization at each node is explicitly captured by its intrinsic random variable $W(\cdot)$, and is assumed to be independent across all nodes of the system.
3) Furthermore, each node is allowed to send a different transmission on each of its outgoing edges.

[10] That is, we assume that the message is a single random variable or vector. It is possible to simultaneously examine the information flows of several (possibly dependent) messages, or of sub-messages within a single message. These cases are examined in Section V-F.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIT.2020.2987806, IEEE Transactions on Information Theory

TABLE I
SUMMARY OF NOTATION

Variable(s)	Meaning
$\mathcal{G}^* = (\mathcal{V}^*, \mathcal{E}^*)$	The original complete directed graph, prior to time-unrolling
$\mathcal{G} = (\mathcal{V}, \mathcal{E})$	The time-unrolled graph making up the computational system
$\mathcal{T}$	The set of all time points, $\{0, 1, \ldots, T\}$
$\mathcal{V}$	The set of all nodes in the computational system
$\mathcal{V}_t$	The subset of nodes at time t
V_t, A_t, B_t, C_t, D_t	A node in the graph at time t
V, A, B, C, D, E	A node in the original complete directed graph $\mathcal{G}^*$, or a node in the computational system at an unspecified time point
$\mathcal{A}, \mathcal{B}$	Some subset of nodes in $\mathcal{V}$
$\mathcal{E}$	The set of all edges in the computational system
$\mathcal{E}_t$	The set† of all edges at time t
$\mathcal{E}'_t$	Some subset‡ of edges in $\mathcal{E}_t$
E_t, P_t, Q_t, R_t, S_t	An edge in the computational system at time t
E, P, Q, R, S	An edge in the original complete directed graph $\mathcal{G}^*$, or an edge in the computational system at an unspecified time point
$X(E_t)$	The random variable representing the transmission on the edge E_t
$X(\{E^{(1)}, E^{(2)}\})$	Short-hand notation for $\{X(E^{(1)}), X(E^{(2)})\}$ (refer Equation (1))
$\mathcal{P}(V_t)$	The set of all incoming edges of V_t $(= \mathcal{V}_{t-1} \times \{V_t\} \subseteq \mathcal{E}_{t-1})$
$\mathbb{Q}(V_t)$	The set of all outgoing edges of V_t $(= \{V_t\} \times \mathcal{V}_{t+1} \subseteq \mathcal{E}_t)$
$W(V_t)$	The intrinsically generated random variable at the node V_t
M	The "message", a random variable that enters the system at time $t = 0$, and whose information flow we seek to understand (refer Definition 3c)
$\mathcal{V}_{\text{ip}}$	The input nodes: the subset of nodes at time 0 whose outgoing transmissions depend on the message M (refer Definition 3c)
f_{V_t}	The function computed by the node V_t (refer Definition 3b)

†Script forms typically denote sets

‡Primed script forms typically denote subsets

4) Note that the condition imposed by Equation (2) introduces dependence between the random variables in the set $X(\mathcal{E})$.
5) For the most part, we will not be concerned with the precise form of the computation being performed by every node. We will only make use of information-theoretic measures applied to the message and to the random variables in the computational system.

Throughout the paper, we use the variables U, V, A, B, C and D to refer to nodes and E, P, Q, R and S to refer to edges. We use their script forms, e.g. $\mathcal{R}$, when referring to sets of nodes and edges, and primed script forms, e.g. $\mathcal{R}'$, when referring to subsets thereof. Once again, the notation we use is summarized in Table I, and depicted in Figure 1 for convenience.

Having defined what we mean by the terms "computational system" and "message", in the following sections we proceed to find a definition for "information flow" and identify properties that this definition satisfies in any computational system.

III. DEFINING INFORMATION FLOW

Before one can speak of *detecting* information flow in a network, it is first important to *define* what it is that we seek to detect.[11] In this section, we focus on arriving at a definition for information flow.

Our goal is to formalize how information about a message flows in a computational system. Ultimately, we expect to find the *path* that the message takes while being processed by the system. Towards this, we start by trying to formally define what it means for information about the message to flow on a given *edge*. This section concludes with a proposal for such a definition: one based on strict positivity of a conditional mutual information. But to provide the intuition behind this choice of definition, we start with several simpler candidate definitions, and show how they fail to satisfy an intuitive property using counterexamples.

After proposing a definition for information flow, in Section IV, we discuss the *properties* satisfied by our definition. Then, in Section V, we specify how the transmissions of the computational system are observed, and describe how information flow might be *inferred* in a real computational system.

A. An intuitive property

To concretely define what it means for information about a message to flow on an edge, we need some way to assess competing candidate definitions and choose one among them. Towards this goal, we state a straightforward and intuitive property, which we would want any definition of information flow to satisfy.

Suppose that, at a given point in time, there is *no* flow of information about the message across *any* edge of a computational system. Note that this includes self-edges, so no node "carries" information about the message within its memory either. Then, we expect that information about the message has ceased to persist in the system, so the information flow about the message *must* be zero on all edges of the computational system, at all future points in time.

Property 1 (The Broken Telephone[12]): Let $\mathcal{C}$ be a computational system, and let $\mathcal{F}_M : \mathcal{E} \to \{0, 1\}$ be an indicator of the presence of information flow about M on an edge. That is, $\mathcal{F}_M(E) = 1$, if information about M flows on the edge $E \in \mathcal{E}$ and $\mathcal{F}_M(E) = 0$, otherwise. The Broken Telephone Property states that if, at some time $t \in \mathcal{T}$, we have

$$\mathcal{F}_M(E_t) = 0 \quad \forall\, E_t \in \mathcal{E}_t, \tag{6}$$

then

$$\mathcal{F}_M(E_{t'}) = 0 \quad \forall\, E_{t'} \in \mathcal{E}_{t'} \;\; \forall\, t' \in \mathcal{T}, t' > t. \tag{7}$$

[11]In essence, "causal influence" measures such as Granger Causality and Directed Information, while intuitively quantifying transferred information, fail to lay down what *aspect of computation* they actually capture. This is, in part, a result of conflating the stages of defining a quantity we want to understand, and prescribing an estimator for it.

[12]https://en.wikipedia.org/wiki/Telephone_game

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIT.2020.2987806, IEEE Transactions on Information Theory

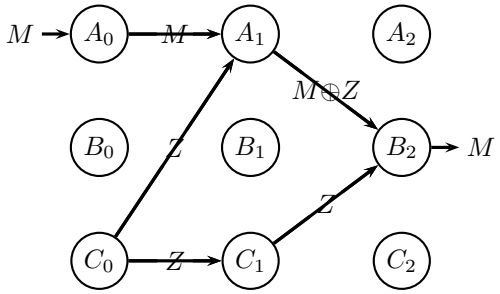


Figure 2: The computational system for Counterexample 1. We only depict edges relevant to the counterexample here. All other edges in the underlying complete directed graph are still present, but are not shown; their transmissions are assumed to be zero. Observe that no edge at time $t = 1$ has information flow as per Candidate Definition 1, yet the message reappears at time $t = 2$.

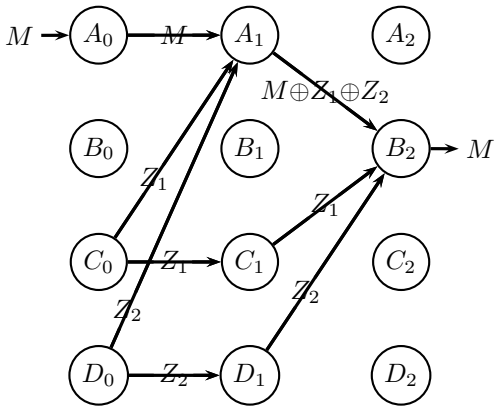


Figure 3: The computational system for Counterexample 2. Once again, observe that no edge at time $t = 1$ has information flow as per Candidate Definition 2, yet the message reappears at time $t = 2$. Note that only edges relevant to the counterexample are depicted in the figure. All other edges of the underlying complete directed graph are still present, and their transmissions are assumed to be zero.

B. Intuiting Information Flow through Counterexamples

We now propose four candidate definitions, beginning with the simplest. We then construct counterexamples to show how the first three candidate definitions do not satisfy Property 1.

Candidate Definition 1: A simplistic and intuitive definition for information flow might simply stem from dependence. We say that information about the message M flows on an edge E_t if

$$I\big(M; X(E_t)\big) > 0.$$

Counterexample 1: Consider the computational system depicted in Figure 2 (note that, in order to avoid unnecessary clutter, only edges with non-zero transmissions are shown in the figure). A_0 is the input node, which has the message $M \sim \text{Ber}(1/2)$ at time $t = 0$. The system is designed to communicate[13] M to the node B using the following strategy: at $t = 0$, A_0 "transmits" M to A_1 (i.e., node A stores M in its memory). C_0 independently generates a different random number, $W(C_0) = Z \sim \text{Ber}(1/2)$, $Z \perp\!\!\!\perp M$, and sends this message to A_1, while also storing it in memory it until $t = 1$. A_1 then computes $M \oplus Z$ and passes the result to B_2, while C_1 sends Z to B_2. Here, the symbol "$\oplus$" stands for XOR, the exclusive-OR operator on two bits. B_2 is thus able to recover M by once again XOR-ing its inputs, $(M \oplus Z)$ and Z.

Note that the output of B_2 depends on M, even though none of its inputs individually depends on M. That is, $I\big(M; X((A_1, B_2))\big) = I(M; M \oplus Z) = 0$, and $I\big(M; X((C_1, B_2))\big) = I(M; Z) = 0$, so by Candidate Definition 1, information about the message flows on *no* edge at time $t = 1$. However, information about the message *does* flow out of node B_2 at time $t = 2$. This violates Property 1. Thus, mere *dependence* on the message cannot be a valid definition for flow of information on a single edge. ■

Communication strategies such as the one in Counterexample 1 frequently arise in cryptography [64], to prevent an eavesdropper from reading confidential information, and in network coding [7], for achieving the communication capacity of a network. Furthermore, a complex computational network may have smaller sub-networks with such topologies. For instance, we observe such a sub-network in the canonical example for network coding: the butterfly network [7, Fig. 7b] (this particular example is discussed in detail in Section VI-A). Optimal communication in such a network *requires* the use of such topologies, so Counterexample 1 is far from obscure. In fact, central to the idea of Counterexample 1 is a concept known as "synergy", which is well-studied in the literature on Partial Information Decomposition [65]–[67] (see [68] for a recent review). This is discussed at length in Section III-E. Even in neuroscience, the concept of synergy is recognized and well-understood [69]–[71], and some experimental evidence has appeared in the literature [72].

Counterexample 1 demonstrates that the information necessary to recover the message (or a function of it) is not necessarily transmitted through individual edges, but jointly across edges. So, we might instead seek to define the "smallest set of edges" along which information about the message flows, for every point in time. But if we ultimately wish to isolate *paths* along which information about the message flows, we require an understanding of which edges *specifically* the information flows upon. We therefore continue to think of information as flowing on individual edges.[14]

We can now update our naïve definition to counter the previous counterexample. We start by noting that in Counterexample 1, although the transmission on edge (A_1, B_2) is independent of M, it is not *conditionally* independent of M when given the transmission on (C_1, B_2).

Candidate Definition 2: We say that information about the message M flows on an edge $E_t \in \mathscr{E}_t$ if at least one of the following holds:

1) $I\big(M; X(E_t)\big) > 0$, or
2) $\exists\, E'_t \in \mathscr{E}_t$ s.t. $I\big(M; X(E_t) \,\big|\, X(E'_t)\big) > 0$.

Counterexample 2: Consider a modified version of Counterexample 1 in which we XOR M with *two* random variables,

[13] This communication can be thought of as computing the identity function, and making the output available at the node B.

[14] It should be noted that the two views—information flowing on individual edges, versus sets of edges—are compatible with each other if we use Definition 5 (which will appear shortly) to describe information flow on a set of edges. This equivalence is elaborated upon in Section III-D. Later, in Section IV-D, we attempt to refine our understanding of the aforementioned "smallest set of edges" along which information about the message flows.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIT.2020.2987806, IEEE Transactions on Information Theory

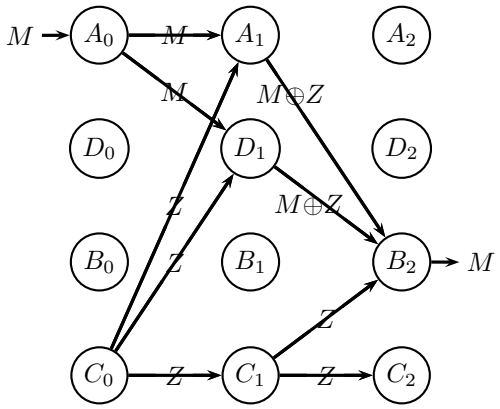


Figure 4: The computational system for Counterexample 3. Just as in the previous counterexamples, no edge at time $t = 1$ has information flow as per Candidate Definition 3, yet the message is reconstructed at time $t = 2$. Note that only edges relevant to the counterexample are depicted in the figure. All other edges of the underlying complete directed graph are still present, and their transmissions are assumed to be zero.

Z_1 and Z_2, where $M, Z_1, Z_2 \sim$ i.i.d. Ber$(1/2)$ (as shown in Figure 3). Now, since there are two noise terms, no single one of them may be conditioned upon to have non-zero information flow at time $t = 1$. That is, $I(M; M \oplus Z_1 \oplus Z_2 \,|\, Z_1) = 0$ and $I(M; M \oplus Z_1 \oplus Z_2 \,|\, Z_2) = 0$. The same holds true of Z_1, conditioned on either $M \oplus Z_1 \oplus Z_2$ or Z_2, and for Z_2, conditioned on either $M \oplus Z_1 \oplus Z_2$ or Z_1. So, Candidate Definition 2 also fails to satisfy Property 1. ■

It might seem that a possible rectification is to condition on *all* other edges at time t, but we can show that this also fails the test.

Candidate Definition 3: We say that information about the message M flows on an edge $E_t \in \mathscr{E}_t$ if at least one of the following holds:

1) $I\big(M; X(E_t)\big) > 0$, or
2) $I\big(M; X(E_t) \,\big|\, X(\mathscr{E}_t \backslash \{E_t\})\big) > 0$.

Counterexample 3: Consider the computational system shown in Figure 4. Once again, we have an input node A_0 which possesses the message at time $t = 0$, and wishes to send this message to node B. It does so by mixing M with an independent random variable Z generated at C_0, so that the scenario described in Counterexample 1 still holds. But additionally, A communicates to B along a redundant path, through D_1. Now, if E is any incoming edge of B_2, it is still true that $I\big(M; X(E)\big) = 0$. So none of the inputs of B_2 individually depends on M, thus eliminating the first condition in Candidate Definition 3. Furthermore, checking each incoming edge of B_2 reveals that the second condition also fails to hold. If we take $E_1 = (A_1, B_2)$, we get

$$I\big(M; X(E_1) \,\big|\, X(\mathscr{E}_1 \backslash \{E_1\})\big) = I(M; M \oplus Z \,|\, M \oplus Z, Z) = 0. \tag{8}$$

The same holds true when $E_1 = (D_1, B_2)$ since the transmissions on both edges are identical by construction. Likewise, if we take $E_1 = (C_1, B_2)$, we have

$$I\big(M; X(E_1) \,\big|\, X(\mathscr{E}_1 \backslash \{E_1\})\big) = I(M; Z \,|\, M \oplus Z, Z) = 0, \tag{9}$$

with the same holding true when $E_1 = (C_1, C_2)$. Therefore, no edge at time $t = 1$ has any information flow about the message M, as per Candidate Definition 3. Nevertheless, B_2 is able to recover the message at time $t = 2$, proving that Property 1 fails to hold for Candidate Definition 3. ■

C. Information Flow on a Single Edge

The counterexamples presented in the previous section motivate a new definition for when information about the message can be said to flow on a given edge. Neither *dependence* of M on the transmission of an edge, nor conditional dependence given *one* or *all* other edges, satisfy Property 1.

However, in all these counterexamples, given an edge E_t upon which we expect to have non-zero information flow, we observe: there is at least one *subset* of edges $\mathscr{E}'_t \subseteq \mathscr{E}_t \setminus \{E_t\}$, such that when given $X(\mathscr{E}'_t)$, $X(E_t)$ is conditionally dependent[15] on M. In Counterexample 1, the edge (A_1, B_2), carrying $M \oplus Z$, is conditionally dependent on M, given $X\big((C_1, B_2)\big) = Z$. In Counterexample 2, $X\big((A_1, B_2)\big) = M \oplus Z_1 \oplus Z_2$ is conditionally dependent on M, given $\{X\big((C_1, B_2)\big), X\big((D_1, B_2)\big)\} = \{Z_1, Z_2\}$. And finally, in Counterexample 3, $X\big((A_1, B_2)\big) = M \oplus Z$ is conditionally dependent on M, given $X\big((C_1, B_2)\big) = Z$; note that we do *not* condition on $X\big((D_1, B_2)\big) = M \oplus Z$. Thus, conditioning on a subset of the other edges' transmissions creates dependence between M and the transmission on an edge of interest.

We will shortly prove that Property 1 holds when information flow is defined as below, so we directly state it as a definition, skipping its candidacy status.

Definition 4 (M-information Flow on a Single Edge): We say that information about the message M flows on an edge $E_t \in \mathscr{E}_t$ if

$$\exists\, \mathscr{E}'_t \subseteq \mathscr{E}_t \setminus \{E_t\} \quad \text{s.t.} \quad I\big(M; X(E_t) \,\big|\, X(\mathscr{E}'_t)\big) > 0. \tag{10}$$

Henceforth, we refer to "information flow about the message M" as *M-information flow*, and use the phrase "the edge E_t *has* M-information flow" or "the edge E_t *carries* M-information flow" to mean that information about M flows on E_t per this definition.

Note that if $I\big(M; X(E_t) \,\big|\, X(\mathscr{E}'_t)\big) > 0$, then we must have $I\big(M; X(\{E_t\} \cup \mathscr{E}'_t)\big) > 0$. In other words, *there exists* a set of edges that includes E_t, whose transmissions depend on M. This is why it is important to condition on all possible subsets of $\mathscr{E}_t$. It is not immediately clear, however, whether *every* edge in $\{E_t\} \cup \mathscr{E}'_t$ has M-information flow. We return to this point in Section IV-D.

Also, this definition implies that certain edges, such as (C_1, B_2) in Counterexample 1, may have M-information flow, which may seem counter-intuitive. This is discussed further and justified in Section IV-B.

D. Information Flow on a Set of Edges

The definition of M-information flow for a single edge naturally generalizes to one for a set of edges, at a given time.

[15]Equivalently, we could say that there exists at least one subset of edges $\mathscr{E}'_t \subseteq \mathscr{E}_t$, without explicitly excluding E_t, since $I\big(M; X(E_t) \,\big|\, X(E_t), X(\mathscr{E}'_t)\big) = 0$.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIT.2020.2987806, IEEE Transactions on Information Theory

Definition 5 (M-information Flow on a Set of Edges): We say that information about the message M flows on a set of edges $\mathscr{E}'_t \subseteq \mathscr{E}_t$ if

$$\exists\, \mathscr{R}'_t \subseteq \mathscr{E}_t \quad \text{s.t.} \quad I\big(M; X(\mathscr{E}'_t) \,\big|\, X(\mathscr{R}'_t)\big) > 0. \tag{11}$$

The definition of M-information flow on a set of edges is nearly identical to its single-edge counterpart. Indeed, they are closely related, as the following proposition shows.

Proposition 1: A set $\mathscr{E}'_t \subseteq \mathscr{E}_t$ has M-information flow (per Definition 5) if and only if there exists an edge $E'_t \in \mathscr{E}'_t$ that has M-information flow (per Definition 4).

A proof of this proposition can be found in Appendix A.

It should be noted that although the counterexamples in this section all employed computational systems which recovered the message M at a new node at a later time, a computational system will in general compute some function of the message. For instance, see the example in Section VI-B.

E. The Connection with Synergistic Information

This section connects our definition of M-information flow with recent developments on a subject known as "Partial Information Decomposition" (PID). Our definition is closely related to the concept of "Synergistic Information" that appears in this field. This section exists only for the purpose of providing a deeper intuition for our definition of M-information flow, and does not affect the rest of the paper in any significant way. We have attempted to explain this intuition in a way that is accessible to readers unfamiliar with the PID literature. However, readers may feel free to skip this section, if desired.

At its core, Counterexample 1 relies on a concept known as "synergy", which is described explicitly in the literature on Partial Information Decomposition (PID) [65]–[67] (see [68] for a recent review, and Appendix C for a brief introduction). Essentially, this body of literature seeks to decompose the mutual information that two or more variables share about a message, $I\big(M; (Y_1, Y_2, \ldots)\big)$, into several individually meaningful, non-negative components. In particular, when discussing the bivariate case—i.e., the case of two variables, $I\big(M; (Y_1, Y_2)\big)$—it is understood what the terms in this decomposition should be: (i) information about the message that each variable carries *uniquely*, and which cannot be inferred from the other; (ii) information about the message that the variables share *redundantly*, and which can be extracted from either; (iii) and information about the message that the variables convey *synergistically*, which is revealed only when *both* variables are taken together, and cannot be inferred from either variable *individually*. Counterexample 1 is the canonical example for synergy, and is known simply as the "XOR" example in the PID literature. While $M \oplus Z$ and Z are *individually* independent of M, when taken *together*, $I\big(M; (M \oplus Z, Z)\big) = H(M)$. This suggests that $M \oplus Z$ and Z have no unique or shared information about M, but convey information synergistically.

While the field has not yet arrived at a consensus on the most appropriate definitions for unique, redundant and synergistic information [68], it is well-understood what properties these quantities must satisfy, at least in the bivariate case (see Appendix C, specifically, Equations (94), (95) and (97)). Therefore, even without formal definitions, we can rely on the intuition provided by these properties to understand the implications of PID for M-information flow. If a particular edge's transmission contains unique or redundant information about the message (with respect to some other subset of edges at that point in time), then that information will manifest itself in the form of strictly positive mutual information. However, in the absence of positive mutual information between the message and the transmission on a given edge, we need to consider whether said transmission synergistically interacts with another subset of transmissions at that point in time, as this could potentially create dependence with the message through the kind of "recombination" described in Counterexample 1. We then need to decide whether such synergistic interactions ought to be considered to constitute information flow. As we show below, our definition of M-information flow *does* consider instances of purely synergistic information to constitute information flow.

Indeed, it is possible to formulate a definition for information flow based on synergy, which is completely equivalent to Definition 4. The definition below makes use of the PID preliminaries given in Appendix C.

Definition 6 (M-synergistic information flow): We say that an edge E_t has *M-synergistic information flow* if at least one of the following holds:

1) $I\big(M; X(E_t)\big) > 0$, or
2) $\exists\, \mathscr{E}'_t \subseteq \mathscr{E}_t \setminus \{E_t\}$ s.t. $SI\big(M : X(E_t); X(\mathscr{E}'_t)\big) > 0$,

where $SI(M : X; Y)$ represents the synergistic information between X and Y about M.

Proposition 2 (Equivalence of Information Flow Definitions): An edge E_t has M-information flow if and only if it has M-synergistic information flow. Furthermore, suppose E_t is an edge which satisfies $I\big(M; X(E_t)\big) = 0$. Then,

$$I\big(M; X(E_t) \,\big|\, X(\mathscr{E}'_t)\big) > 0 \tag{12}$$

for some set $\mathscr{E}'_t \subseteq \mathscr{E}_t \setminus \{E_t\}$, if and only if

$$SI\big(M : X(E_t); X(\mathscr{E}'_t)\big) > 0. \tag{13}$$

That is, the set $\mathscr{E}'_t$ upon whose transmissions we need to condition is the *same* as the one responsible for providing synergy in the alternate definition.

A proof of this proposition is given in Appendix C.

We should also mention here that it may be possible to leverage specific definitions of synergistic information to supply an intuitive measure of the *volume* of information flow; we discuss this in Section VII-E.

IV. Properties of Information Flow

Having defined what it means for information about a message to flow on an edge, we demonstrate that Definition 4 satisfies several intuitively desirable properties, including Property 1.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIT.2020.2987806, IEEE Transactions on Information Theory

A. The Broken Telephone Property

Theorem 3: M-information flow satisfies the Broken Telephone Property, i.e., Definition 4 satisfies Property 1.

Before we prove this theorem, we prove a simpler lemma which directly falls out of Definition 4 and the properties of mutual information.

Lemma 4: There is *no* edge in $\mathscr{E}_t$ that carries M-information flow if, and only if, $X(\mathscr{E}_t)$ is independent of M. In other words,

$$I\big(M; X(E_t) \,\big|\, X(\mathscr{E}'_t)\big) = 0 \quad \forall\, E_t \in \mathscr{E}_t,\ \mathscr{E}'_t \subseteq \mathscr{E}_t \backslash \{E_t\} \tag{14}$$

if and only if

$$I\big(M; X(\mathscr{E}_t)\big) = 0. \tag{15}$$

Equivalently, we can state the opposite: $X(\mathscr{E}_t)$ *depends* on M if and only if *at least* one edge in $\mathscr{E}_t$ carries M-information flow.

Proof: ($\Rightarrow$) Suppose that the condition in (14) holds. Let $\mathscr{E}_t = \big\{E_t^{(1)}, E_t^{(2)}, \ldots, E_t^{(N^2)}\big\}$ be any ordering of the edges in $\mathscr{E}_t$. Then,

$$\begin{aligned} &I\big(M; X(\mathscr{E}_t)\big) \\ &\quad \overset{(a)}{=} I\big(M; X(E_t^{(1)})\big) + I\big(M; X(E_t^{(2)}) \,\big|\, X(E_t^{(1)})\big) && (16)\\ &\qquad + I\big(M; X(E_t^{(3)}) \,\big|\, X(E_t^{(1)}), X(E_t^{(2)})\big) + \cdots \\ &\quad = \sum_{i=1}^{N^2} I\Big(M; X(E_t^{(i)}) \,\Big|\, \bigcup_{j=1}^{i-1} \big\{X(E_t^{(j)})\big\}\Big) && (17)\\ &\quad \overset{(b)}{=} \sum_{i=1}^{N^2} I\Big(M; X(E_t^{(i)}) \,\Big|\, X\Big(\bigcup_{j=1}^{i-1} \{E_t^{(j)}\}\Big)\Big) \overset{(c)}{=} 0, && (18) \end{aligned}$$

where (a) follows from the chain-rule of mutual information [73, Ch. 2], (b) is simply the application of Equation (1), and (c) follows from the fact that each term in the summation is zero, by (14). This proves the forward implication.

($\Leftarrow$) Next, suppose $I\big(M; X(\mathscr{E}_t)\big) = 0$. Let E_t be any edge in $\mathscr{E}_t$ and let $\mathscr{E}'_t$ be any subset of $\mathscr{E}_t \setminus \{E_t\}$. Also, let $\mathscr{E}''_t = \mathscr{E}_t \setminus \big(\mathscr{E}'_t \cup \{E_t\}\big)$. Then,

$$\begin{aligned} 0 &= I\big(M; X(\mathscr{E}_t)\big) && (19)\\ &= I\big(M; X(\mathscr{E}'_t)\big) + I\big(M; X(E_t) \,\big|\, X(\mathscr{E}'_t)\big) \\ &\quad + I\big(M; X(\mathscr{E}''_t) \,\big|\, X(\mathscr{E}'_t), X(E_t)\big) && (20) \end{aligned}$$

by the chain rule. Since (conditional) mutual information is always non-negative [73, Ch. 2], all three terms on the right hand side must be zero. So in particular,

$$I\big(M; X(E_t) \,\big|\, X(\mathscr{E}'_t)\big) = 0. \tag{21}$$

Since E_t and $\mathscr{E}'_t$ are arbitrary, this proves the converse. ∎

Proof of Theorem 3: We need to prove that M-information flow, as given by Definition 4, satisfies Property 1. Explicitly stated, we need to show that if every edge at some time t has zero M-information flow, then every edge at all future times $t' > t$ must also have zero M-information flow. So suppose that, at time t, for every $E_t \in \mathscr{E}_t$ we have

$$I\big(M; X(E_t) \,\big|\, X(\mathscr{E}'_t)\big) = 0 \quad \forall\, \mathscr{E}'_t \subseteq \mathscr{E}_t \setminus \{E_t\}. \tag{22}$$

By Lemma 4, this implies that

$$I\big(M; X(\mathscr{E}_t)\big) = 0. \tag{23}$$

Now, consider the first future time instant, $t' = t+1$. For every node $A_{t+1} \in \mathscr{V}_{t+1}$, the definition of computation at a node (Definition 3b) states that

$$X(\mathbb{Q}(A_{t+1})) = f_{A_{t+1}}\big(X(\mathscr{P}(A_{t+1})), W(A_{t+1})\big), \tag{24}$$

where the reader may recall, $\mathscr{P}(A_{t+1})$ and $\mathbb{Q}(A_{t+1})$ are the edges entering and leaving A_{t+1} respectively. We can collect the individual functions $f_{A_{t+1}}$ across all nodes in $\mathscr{V}_{t+1}$ into a single joint function $f_{\mathscr{V}_{t+1}}$, as described in Definition 3b, to obtain

$$X(\mathscr{E}_{t+1}) = f_{\mathscr{V}_{t+1}}\big(X(\mathscr{E}_t), W(\mathscr{V}_{t+1})\big). \tag{25}$$

Therefore,

$$\begin{aligned} 0 &\overset{(a)}{\leq} I\big(M; X(\mathscr{E}_{t+1})\big) \\ &= I\big(M; f_{\mathscr{V}_{t+1}}\big(X(\mathscr{E}_t), W(\mathscr{V}_{t+1})\big)\big) && (26)\\ &\overset{(b)}{\leq} I\big(M; X(\mathscr{E}_t), W(\mathscr{V}_{t+1})\big) && (27)\\ &= I\big(M; X(\mathscr{E}_t)\big) + I\big(M; W(\mathscr{V}_{t+1}) \,\big|\, X(\mathscr{E}_t)\big) && (28)\\ &\overset{(c)}{=} I\big(M; X(\mathscr{E}_t)\big) \overset{(d)}{=} 0, && (29) \end{aligned}$$

where (a) follows from the non-negativity of mutual information, (b) is an application of the Data Processing Inequality [73, Ch. 2], (c) follows from the fact that $W(\mathscr{V}_{t+1}) \perp\!\!\!\perp \{M, X(\mathscr{E}_t)\}$, as stated in Definition 3b, and (d) follows from (23). So, we must have that $I\big(M; X(\mathscr{E}_{t+1})\big) = 0$. Applying Lemma 4 once again, we find that for $t' = t+1$,

$$I\big(M; X(E_{t'}) \,\big|\, X(\mathscr{E}'_{t'})\big) = 0 \quad \forall\, E_{t'} \in \mathscr{E}_{t'}, \mathscr{E}'_{t'} \subseteq \mathscr{E}_{t'} \backslash \{E_{t'}\} \tag{30}$$

We have shown that (22) implies (30), so induction on t' yields that (30) holds for all future times $t' > t$, completing the proof. ∎

B. The Existence of Orphans

M-information flow (Definition 4) also has a very non-intuitive property: an edge leading out of a node may have M-information flow, even though *no* edge leading *into* that node has M-information flow.

Definition 7 (M-information Orphan): In a computational system $\mathscr{C}$, a node V_t is said to be an *M-information orphan* if $\mathbb{Q}(V_t)$ has M-information flow (as per Definition 5), but $\mathscr{P}(V_t)$ has no M-information flow.

Property 2: M-information orphans may exist in a computational system.

Proof: Consider the computational system in Figure 2 from Counterexample 1. The node C_1 is an M-information orphan, since its outgoing edge (C_1, B_2) carries M-information flow, whereas none of its incoming edges carries M-information flow. ∎

The existence of M-information orphans, along with the presence of M-information flow on (C_1, B_2) in Counterexample 1, may not be expected, since Z was never computed from M. Indeed, M-information flow appears to emerge from

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIT.2020.2987806, IEEE Transactions on Information Theory

"nowhere" at the node C_1, leaving it *orphaned* in a view of the graph that contains only edges having M-information flow (hence the name). But closer inspection reveals that in this example, the transmissions arriving at B_2 from A_1 and C_1, i.e. $M \oplus Z$ and Z, are *statistically identical*: they are both individually independent of M, but when XOR'ed, are fully dependent on M. In other words, any *purely observational* measure[16] defined on the transmissions at time t that assigns M-information flow to $M \oplus Z$, must also assign M-information flow to Z.

Note that, just as M-information flow can originate at an M-information orphan, M-information flow may also terminate at a node—either by simple omission, or as a result of some computation (see Section VI for such instances). Likewise, multiple outgoing edges of a given node may transmit redundant copies of the same information. Ultimately, we see that there is no "law of conservation" for M-information flow. In this sense, "information flow" is not a typical kind of "flow" that is defined on graphs (see, for example, [74, Sec. 26.1]), and well-known results such as the Max-flow Min-cut Theorem [74, Thm. 26.6] do not apply as-is to M-information flow.

It is worthwhile to note at this point that the existence of M-information orphans such as C_1 in Counterexample 1 is not inconsistent with the Data Processing Inequality [73, Ch. 2]. In fact, a clear example of the Data Processing Inequality is seen at the network-level, wherein M—$X(\mathscr{E}_t)$—$X(\mathscr{E}_{t+1})$ form a Markov Chain for any time $0 \leq t < T$, and so the information content about M present collectively in all transmissions at time $t+1$ *must* be no more than that present at time t. We call this Global Markovity, and state it formally for completeness.

Corollary 5 (Global Markovity): At any given time t, the following Markov Chain holds: M—$X(\mathscr{E}_t)$—$X(\mathscr{E}_{t+1})$.

In fact, this Markov condition must hold for every *subset* of nodes, not just for the entire set of nodes, so it is subsumed by the following proposition.

Proposition 6 (Local Markovity): At any time t, for any given subset of nodes $\mathscr{V}_t' \subseteq \mathscr{V}_t$, the following Markov Chain holds: M—$X(\mathscr{P}(\mathscr{V}_t'))$—$X(\mathbb{Q}(\mathscr{V}_t'))$.

Proof: Since $X(\mathbb{Q}(\mathscr{V}_t')) = f_{\mathscr{V}_t'}\big(X(\mathscr{P}(\mathscr{V}_t')), W(\mathscr{V}_t')\big)$ by Definition 3b, the tuple $\big(X(\mathscr{P}(\mathscr{V}_t')), X(\mathbb{Q}(\mathscr{V}_t'))\big)$ is also a function of $X(\mathscr{P}(\mathscr{V}_t'))$ and $X(W(\mathscr{V}_t'))$. Hence, the following Markov chain holds:

$$M\text{—}\big(X(\mathscr{P}(\mathscr{V}_t')), W(\mathscr{V}_t')\big)\text{—}\big(X(\mathscr{P}(\mathscr{V}_t')), X(\mathbb{Q}(\mathscr{V}_t'))\big).$$

By the Data Processing Inequality, this implies that

$$\begin{aligned} &I\big(M; X(\mathbb{Q}(\mathscr{V}_t')), X(\mathscr{P}(\mathscr{V}_t'))\big) \\ &\quad\leq I\big(M; X(\mathscr{P}(\mathscr{V}_t')), W(\mathscr{V}_t')\big) && (31)\\ &\quad\overset{(a)}{=} I\big(M; X(\mathscr{P}(\mathscr{V}_t'))\big) + I\big(M; W(\mathscr{V}_t') \,\big|\, X(\mathscr{P}(\mathscr{V}_t'))\big) && (32)\\ &\quad\overset{(b)}{=} I\big(M; X(\mathscr{P}(\mathscr{V}_t'))\big) + I\big(W(\mathscr{V}_t'); M, X(\mathscr{P}(\mathscr{V}_t'))\big) && (33)\\ &\qquad - I\big(W(\mathscr{V}_t'); X(\mathscr{P}(\mathscr{V}_t'))\big) \\ &\quad\overset{(c)}{=} I\big(M; X(\mathscr{P}(\mathscr{V}_t'))\big) + 0 - 0, && (34)\end{aligned}$$

where in (a) and (b), we have used the chain rule of mutual information in two different ways, and in (c) we have used the fact that $W(\mathscr{V}_t') \perp\!\!\!\perp \{M, X(\mathscr{P}(\mathscr{V}_t'))\}$. Therefore,

$$I\big(M; X(\mathbb{Q}(\mathscr{V}_t')) \,\big|\, X(\mathscr{P}(\mathscr{V}_t'))\big) = 0, \qquad (35)$$

which implies the Markov chain in Proposition 6. ∎

Since the above also holds for $\mathscr{V}_t' = \mathscr{V}_t$, wherein $\mathbb{Q}(\mathscr{V}_t) = \mathscr{E}_t$, Proposition 6 implies Corollary 5.

Given that these Markov conditions arise directly from the way we have defined the computational system, specifically Definition 3b, they may not be very surprising (indeed, they may be considered *properties* of the computational system model itself). However, it is worth noting that Proposition 6 holds *even at an M-information orphan*. Thus, M-information orphans do not "create" information about M, as we would rightly expect, given the Data Processing Inequality.

C. The Existence of Information Paths

We now show that if the outgoing transmissions of any given node depend on the message, then we can find a path leading to that node from one or more input nodes, along which M-information flows. Before we demonstrate this property, we formally define what we mean by the terms "path" and "cut".

Definition 8 (Path): In any computational system $\mathscr{C}$, suppose $\mathscr{A}$ and $\mathscr{B}$ are two disjoint sets of nodes in $\mathscr{V}$. Then, a *path* from $\mathscr{A}$ to $\mathscr{B}$ is any ordered set of nodes $\{V^{(0)}, V^{(1)}, \ldots, V^{(L)}\}$ that satisfies (i) $V^{(0)} \in \mathscr{A}$; (ii) $V^{(L)} \in \mathscr{B}$; and (iii) $(V^{(i-1)}, V^{(i)}) \in \mathscr{E}$ for every $1 \leq i \leq L$, where L is a positive integer indicating the length of the path. We refer to the set $\{(V^{(i-1)}, V^{(i)})\}_{i=1}^{L}$ as the *edges of the path*.

Definition 9 (M-Information Path): Continuing from Definition 8, we define an *M-information path* from $\mathscr{A}$ to $\mathscr{B}$ as any path from $\mathscr{A}$ to $\mathscr{B}$, each of whose edges carries M-information flow. That is, if $(V^{(i-1)}, V^{(i)}) = E_{t_i} \in \mathscr{E}_{t_i}$ for some $t_i \in \mathscr{T}$, then for every $1 \leq i \leq L$,

$$\exists\, \mathscr{E}_{t_i}' \subseteq \mathscr{E}_{t_i} \quad \text{s.t.} \quad I\big(M; X(E_{t_i}) \,\big|\, X(\mathscr{E}_{t_i}')\big) > 0. \qquad (36)$$

Definition 10 (Cut): In any computational system $\mathscr{C}$, suppose $\mathscr{A}$ and $\mathscr{B}$ are two disjoint sets of nodes in $\mathscr{V}$. Then, a *cut* separating $\mathscr{A}$ and $\mathscr{B}$ is any pair of sets $(\mathscr{V}^{\text{src}}, \mathscr{V}^{\text{sink}})$, such that (i) $\mathscr{V}^{\text{src}} \cup \mathscr{V}^{\text{sink}} = \mathscr{V}$; (ii) $\mathscr{V}^{\text{src}} \cap \mathscr{V}^{\text{sink}} = \emptyset$; (iii) $\mathscr{A} \subseteq \mathscr{V}^{\text{src}}$; and (iv) $\mathscr{B} \subseteq \mathscr{V}^{\text{sink}}$. We refer to the set of edges *going from $\mathscr{V}^{src}$ to $\mathscr{V}^{sink}$*, i.e. $\mathscr{E} \cap (\mathscr{V}^{\text{src}} \times \mathscr{V}^{\text{sink}})$, as the *edges in the cut set*.[17]

Definition 11 (Zero–M-information Cut): Continuing from Definition 10, we say that a cut $(\mathscr{V}^{\text{src}}, \mathscr{V}^{\text{sink}})$ is a *zero–M-information cut* if every edge in its cut set has zero M-information flow. That is, for every $E_t \in \mathscr{E} \cap (\mathscr{V}^{\text{src}} \times \mathscr{V}^{\text{sink}})$,

$$I\big(M; X(E_t) \,\big|\, X(\mathscr{E}_t')\big) = 0 \quad \forall\, \mathscr{E}_t' \subseteq \mathscr{E}_t \setminus \{E_t\}. \qquad (37)$$

Remark: In Definition 11, we require that Equation (37) hold for every edge E_t in $\mathscr{E} \cap (\mathscr{V}^{\text{src}} \times \mathscr{V}^{\text{sink}})$. However, the

[16] i.e., a functional of the joint distribution of $X(\mathscr{E}_t)$

[17] Note that it is not necessary for us to assume that, individually, $\mathscr{V}^{\text{src}}$ and $\mathscr{V}^{\text{sink}}$ are *connected* sets of nodes. For instance, there may be an isolated subset of $\mathscr{V}^{\text{sink}}$, surrounded only by nodes in $\mathscr{V}^{\text{src}}$. Our theorems and proofs remain unaffected, even in such a scenario.

edges in this set may belong to several different time instants, since the cut is not restricted to any particular time (e.g., see Figure 5). The time t used in Equation (37), therefore, is determined by the time of the edge E_t, and varies for each E_t that we check in $\mathscr{E} \cap (\mathscr{V}^{\text{src}} \times \mathscr{V}^{\text{sink}})$.

Property 3 (Existence of an Information Path): In any computational system $\mathscr{C}$, suppose that at some time $t_{\text{op}} \in \mathscr{T}$, there is an "output node" $V_{\text{op}} \in \mathscr{V}$ whose outgoing edges $\mathbb{Q}(V_{\text{op}})$ satisfy $I\big(M; X(\mathbb{Q}(V_{\text{op}}))\big) > 0$. Then, there must exist an M-information path from the input nodes $\mathscr{V}_{\text{ip}}$ to V_{op}.

Theorem 7: Definition 4 satisfies Property 3.

Informally put, Theorem 7 states that our definition of M-information flow (Definition 4) guarantees M-information paths to every output node whose outgoing transmissions depend on the message. While the theorem seems obvious on the surface, the proof is in fact non-trivial because of the nature of our definition of M-information flow. Due to Property 2, M-information flowing *out of* a node does *not* imply that M-information must flow *into* that node. Therefore, a straightforward application of the Data Processing Inequality at every node fails to prove the theorem, and we must resort to a more rigorous cut-set-based approach.

Proof outline: We shall prove the contrapositive of the theorem, i.e., we will show that if there exists no M-information path from $\mathscr{V}_{\text{ip}}$ to V_{op}, then the outgoing transmissions of V_{op} are independent of M. We first connect the absence of any M-information path with the presence of a zero–M-information cut. This is achieved in Lemma 8, which we present before the proof of Theorem 7.

The proof itself proceeds by induction over time. We divide the proof into two steps: initialization and continuation. Starting with the first nodes that come after the cut (temporally) in the initialization step, we systematically show that all nodes to the right of the cut have outgoing transmissions that are independent of the message M through induction. In this proof outline, we show these steps intuitively using Figure 5, where the dashed black line denotes the cut.

Initialization. Here, node C_1 is the first node to the right of the cut, and all of its incoming edges must come from across the cut (depicted by lines in red). Because the cut is a zero–M-information cut, none of its incoming transmissions have M-information flow. Furthermore, the intrinsically generated random variable $W(C_1)$ is independent of M and all past transmissions. Using these two facts along with the Data Processing Inequality, we can show that the transmissions on C_1's outgoing edges, $X(\mathbb{Q}(C_1))$, are also independent of M.

Continuation. At the second time instant to the right of the cut, nodes B_2 and C_2 receive their incoming transmissions from either C_1 (shown in orange) or from across the cut (shown in blue). Once again, the transmissions coming from across the cut can have no information flow, and we have shown that the transmissions coming from C_1 are independent of M. Also, $W(B_2)$ and $W(C_2)$ are independent of M and all incoming transmissions. This suffices to show that the outgoing transmissions of B_2 and C_2, $X\big(\mathbb{Q}(B_2) \cup \mathbb{Q}(C_2)\big)$, are independent of M. Applying this argument repeatedly over time shows that the transmissions of all nodes to the right of the cut are independent of M.

Therefore, if there is a node V_{op} whose outputs depend on M, we can be assured that there exists no zero–M-information cut separating $\mathscr{V}_{\text{ip}}$ from V_{op}. Therefore, by Lemma 8, there exists an M-information path from $\mathscr{V}_{\text{ip}}$ to V_{op}. □

A few nuances are omitted in this outline, such as how the definition of $\mathscr{V}_{\text{ip}}$ plays a role precisely. These subtleties are better elucidated in the full proof.

Before proceeding to the formal proof of Theorem 7, we first state and prove the lemma we alluded to earlier, which shows how the absence of an M-information path implies the presence of a zero–M-information cut, and vice versa.

Lemma 8: Let $\mathscr{A}$ and $\mathscr{B}$ be two disjoint sets of nodes in the computational system $\mathscr{C}$. There exists no M-information path from $\mathscr{A}$ to $\mathscr{B}$ if and only if there is a zero–M-information cut separating $\mathscr{A}$ and $\mathscr{B}$.

Proof: ($\Rightarrow$) Suppose there exists no M-information path from $\mathscr{A}$ to $\mathscr{B}$. Consider the set of all nodes to which there exists at least one M-information path from $\mathscr{A}$. Let $\mathscr{V}^{\text{src}}$ be the collection of all such nodes, along with the nodes in $\mathscr{A}$, i.e.,

$$\mathscr{V}^{\text{src}} \coloneqq \mathscr{A} \cup \{V_t \in \mathscr{V} : \exists \text{ an } M\text{-information path from } \mathscr{A} \text{ to } V_t\}. \tag{38}$$

Let $\mathscr{V}^{\text{sink}} = \mathscr{V} \setminus \mathscr{V}^{\text{src}}$, so that $\mathscr{V}^{\text{sink}}$ consists of nodes to which there is no M-information path from $\mathscr{A}$. Then, we must have $\mathscr{B} \subseteq \mathscr{V}^{\text{sink}}$, since it is known that there are no M-information paths from $\mathscr{A}$ to $\mathscr{B}$. Therefore, $(\mathscr{V}^{\text{src}}, \mathscr{V}^{\text{sink}})$ is a cut that separates $\mathscr{A}$ and $\mathscr{B}$, such that no edge in the cut set has M-information flow. In other words, by Definition 11, this is a zero–M-information cut separating $\mathscr{A}$ and $\mathscr{B}$.

($\Leftarrow$) Next, suppose that there *is* an M-information path $\{V^{(i)}\}_{i=0}^{L}$ from $\mathscr{A}$ to $\mathscr{B}$. Then, we claim that there can exist no zero–M-information cut separating $\mathscr{A}$ and $\mathscr{B}$. Let $(\mathscr{V}^{\text{src}}, \mathscr{V}^{\text{sink}})$ be any cut separating $\mathscr{A}$ and $\mathscr{B}$. By Definitions 8 and 10, we must have $V^{(0)} \in \mathscr{A} \subseteq \mathscr{V}^{\text{src}}$ and $V^{(L)} \in \mathscr{B} \subseteq \mathscr{V}^{\text{sink}}$. So, there must be at least one edge going from $\mathscr{V}^{\text{src}}$ to $\mathscr{V}^{\text{sink}}$ which lies on the path. This implies that at least one edge in the cut set carries M-information flow. Since the conditions of Definition 11 are not satisfied, this cut is *not* a zero–M-information cut. Since this is true for every cut separating $\mathscr{A}$ and $\mathscr{B}$, the claim holds. ■

Proof of Theorem 7: As mentioned in the proof outline, we prove the contrapositive of the theorem. Suppose there exists no M-information path from the input nodes $\mathscr{V}_{\text{ip}}$ to V_{op}. Then, by Lemma 8, there exists a zero–M-information cut[18] separating $\mathscr{V}_{\text{ip}}$ and V_{op}. We use this to prove that the transmissions of V_{op} are independent of M.

Setup. Let the zero–M-information cut separating $\mathscr{V}_{\text{ip}}$ and V_{op} be given by $(\mathscr{V}^{\text{src}}, \mathscr{V}^{\text{sink}})$, so that $\mathscr{V}_{\text{ip}} \subseteq \mathscr{V}^{\text{src}}$ and $V_{\text{op}} \in \mathscr{V}^{\text{sink}}$. Then, the cut divides $\mathscr{E}$ into the following sets:

1) $\mathscr{E}^{\text{src}} \coloneqq \mathscr{E} \cap (\mathscr{V}^{\text{src}} \times \mathscr{V}^{\text{src}})$, the edges between the nodes in $\mathscr{V}^{\text{src}}$;
2) $\mathscr{E}^{\text{sink}} \coloneqq \mathscr{E} \cap (\mathscr{V}^{\text{sink}} \times \mathscr{V}^{\text{sink}})$, the edges between the nodes in $\mathscr{V}^{\text{sink}}$; and

[18]Note that, in general, this cut may be arbitrarily complex, spanning several nodes and multiple time instants.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIT.2020.2987806, IEEE Transactions on Information Theory

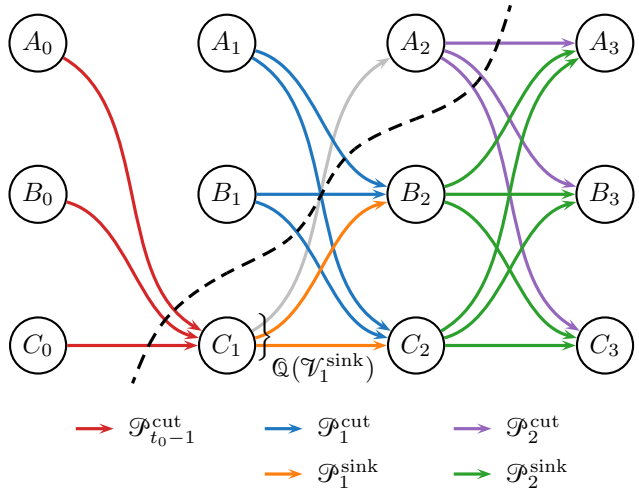


Figure 5: A generic computational system used in the proof outline and to explain certain steps in the proof of Theorem 7. For the purposes of the proof outline, it suffices to note that the black dashed line denotes the cut. All variable names can be ignored at this point of time.
For the purposes of the formal proof, note that in this figure, $\mathcal{E}^{\text{cut}}$ is essentially the union of the red, blue and purple edges, while $\mathcal{E}^{\text{sink}}$ is the union of the orange and green edges. From this, it is evident that $\mathcal{P}(\mathcal{V}_t^{\text{sink}}) = \mathcal{P}_{t-1}^{\text{cut}} \cup \mathcal{P}_{t-1}^{\text{sink}}$ for any time t, i.e., the incoming edges of $\mathcal{V}^{\text{sink}}$ at time t must either come from nodes in $\mathcal{V}^{\text{sink}}$ or from nodes across the cut. Secondly, it should be clear that $\mathcal{P}_{t-1}^{\text{sink}} = \mathbb{Q}(\mathcal{V}_{t-1}^{\text{sink}}) \cap \mathcal{E}^{\text{sink}}$, i.e., the *incoming* edges of $\mathcal{V}_t^{\text{sink}}$ that originate from nodes in $\mathcal{V}^{\text{sink}}$ are simply the *outgoing* edges of $\mathcal{V}_{t-1}^{\text{sink}}$ which terminate at nodes in $\mathcal{V}^{\text{sink}}$. This is seen best at time $t = 1$ in the graph above, where the orange and grey lines together represent $\mathbb{Q}(\mathcal{V}_1^{\text{sink}})$, the orange and green edges together make up $\mathcal{E}^{\text{sink}}$, and $\mathcal{P}_1^{\text{sink}}$ is given by the orange edges, which is the intersection of the two sets.

3) $\mathcal{E}^{\text{cut}} := \mathcal{E} \cap (\mathcal{V}^{\text{src}} \times \mathcal{V}^{\text{sink}})$, the edges going from $\mathcal{V}^{\text{src}}$ to $\mathcal{V}^{\text{sink}}$.

(Note that the edges going from $\mathcal{V}^{\text{sink}}$ to $\mathcal{V}^{\text{src}}$ will not be relevant to our discussion). As stated before, Lemma 8 implies that $(\mathcal{V}^{\text{src}}, \mathcal{V}^{\text{sink}})$ is a zero–M-information cut, so by Definition 11, we have that for all $E_t \in \mathcal{E}^{\text{cut}}$,

$$I\big(M; X(E_t) \,\big|\, X(\mathcal{E}_t')\big) = 0 \quad \forall\, \mathcal{E}_t' \subseteq \mathcal{E}_t \setminus \{E_t\}. \tag{39}$$

Note that the edges in $\mathcal{E}^{\text{cut}}$ may belong to different time instants. In particular, the time instant t in the equation above corresponds to the time of the edge E_t, whose flow is in question.[19]

Order the nodes in $\mathcal{V}^{\text{sink}}$ by time, and let $\mathcal{V}_t^{\text{sink}}$ be the subset of nodes in $\mathcal{V}^{\text{sink}}$ at time t. Let $\mathcal{P}(\mathcal{V}_t^{\text{sink}})$ and $\mathbb{Q}(\mathcal{V}_t^{\text{sink}})$ respectively be the sets of edges *collectively* entering and leaving all nodes in $\mathcal{V}_t^{\text{sink}}$. We shall prove that the outgoing transmissions of every node in $\mathcal{V}^{\text{sink}}$, including those of V_{op}, must be independent of the message, i.e.,

$$I\big(M; X(\mathbb{Q}(V))\big) = 0 \quad \forall\, V \in \mathcal{V}^{\text{sink}}. \tag{40}$$

Initialization. Let t_0 be the first time instant t for which $\mathcal{V}_t^{\text{sink}}$ is non-empty. Then, we encounter two cases: either $t_0 = 0$, in which case the nodes in $\mathcal{V}_{t_0}^{\text{sink}}$ have *no* incoming edges, or $t_0 > 0$, and the nodes in $\mathcal{V}_{t_0}^{\text{sink}}$ *have* incoming edges. We shall first prove that in *both* cases, the outgoing transmissions of $\mathcal{V}_{t_0}^{\text{sink}}$ are independent of the message, i.e. $I\big(M; X(\mathbb{Q}(\mathcal{V}_{t_0}^{\text{sink}}))\big) = 0$.

(Case I) When $t_0 = 0$, $\mathcal{V}_0^{\text{sink}} \cap \mathcal{V}_{\text{ip}} = \emptyset$. This is because the cut separates $\mathcal{V}_{\text{ip}}$ from V_{op}, with $\mathcal{V}_{\text{ip}} \subseteq \mathcal{V}^{\text{src}}$, so no nodes in $\mathcal{V}_0^{\text{sink}}$ can be input nodes. So, by the definition of (non-)input nodes (Definition 3c), we must have

$$I\big(M; X(\mathbb{Q}(\mathcal{V}_0^{\text{sink}}))\big) = I\big(M; f_{\mathcal{V}_0^{\text{sink}}}(W(\mathcal{V}_0^{\text{sink}}))\big) \tag{41}$$

$$\overset{(a)}{\leq} I\big(M; W(\mathcal{V}_0^{\text{sink}})\big) \tag{42}$$

$$\overset{(b)}{=} 0, \tag{43}$$

where step (a) uses the data processing inequality and step (b) makes use of the fact that $W(\mathcal{V}_0) \perp\!\!\!\perp M$.

(Case II) When $t_0 > 0$, the definition of t_0 implies that all nodes at time $t_0 - 1$ are in $\mathcal{V}^{\text{src}}$, so all incoming edges of $\mathcal{V}_{t_0}^{\text{sink}}$ must lie in the cut set, i.e., $\mathcal{P}(\mathcal{V}_{t_0}^{\text{sink}}) \subseteq \mathcal{E}^{\text{cut}}$. Since the cut is a zero–M-information cut, we have that for all $E_{t_0-1} \in \mathcal{P}(\mathcal{V}_{t_0}^{\text{sink}})$,

$$I\big(M; X(E_{t_0-1}) \,\big|\, X(\mathcal{E}_{t_0-1}')\big) = 0 \quad \forall\, \mathcal{E}_{t_0-1}' \subseteq \mathcal{E}_{t_0-1}. \tag{44}$$

By the definition of M-information flow for a set of edges (Definition 5) and Proposition 1, we have

$$I\big(M; X(\mathcal{P}(\mathcal{V}_{t_0}^{\text{sink}})) \,\big|\, X(\mathcal{E}_{t_0-1}')\big) = 0 \quad \forall\, \mathcal{E}_{t_0-1}' \subseteq \mathcal{E}_{t_0-1}. \tag{45}$$

Once again, considering $\mathbb{Q}(\mathcal{V}_{t_0}^{\text{sink}})$, we have

$$I\big(M; X(\mathbb{Q}(\mathcal{V}_{t_0}^{\text{sink}}))\big)$$

$$= I\big(M; f_{\mathcal{V}_{t_0}^{\text{sink}}}\big(X(\mathcal{P}(\mathcal{V}_{t_0}^{\text{sink}})), W(\mathcal{V}_{t_0}^{\text{sink}})\big)\big) \tag{46}$$

$$\overset{(a)}{\leq} I\big(M; X(\mathcal{P}(\mathcal{V}_{t_0}^{\text{sink}})), W(\mathcal{V}_{t_0}^{\text{sink}})\big) \tag{47}$$

$$\overset{(b)}{=} I\big(M; X(\mathcal{P}(\mathcal{V}_{t_0}^{\text{sink}}))\big) + I\big(M; W(\mathcal{V}_{t_0}^{\text{sink}}) \,\big|\, X(\mathcal{P}(\mathcal{V}_{t_0}^{\text{sink}}))\big) \tag{48}$$

$$\overset{(c)}{=} 0, \tag{49}$$

where (a) and (b) follow from the Data Processing Inequality and the chain rule of mutual information respectively. In step (c), the first expression in the sum goes to zero by taking $\mathcal{E}_{t_0-1} = \emptyset$ in (45) and the second expression is zero since $W(\mathcal{V}_{t_0}^{\text{sink}}) \perp\!\!\!\perp \{M, X(\mathcal{E}_{t_0-1})\}$, and $\mathcal{P}(\mathcal{V}_{t_0}^{\text{sink}}) \subseteq \mathcal{E}_{t_0-1}$ (refer Definition 3b). So, from equations (43) and (49), we have that for all values of t_0,

$$I\big(M; X(\mathbb{Q}(\mathcal{V}_{t_0}^{\text{sink}}))\big) = 0. \tag{50}$$

Continuation. Now, suppose that for some $t > t_0$, we have $I\big(M; X(\mathbb{Q}(\mathcal{V}_{t-1}^{\text{sink}}))\big) = 0$. We shall prove that this implies $I\big(M; X(\mathbb{Q}(\mathcal{V}_t^{\text{sink}}))\big) = 0$. First, observe that

$$\mathcal{P}(\mathcal{V}_t^{\text{sink}}) = (\mathcal{P}(\mathcal{V}_t^{\text{sink}}) \cap \mathcal{E}^{\text{cut}}) \cup (\mathcal{P}(\mathcal{V}_t^{\text{sink}}) \cap \mathcal{E}^{\text{sink}}) \tag{51}$$

For convenience, let $\mathcal{P}_{t-1}^{\text{cut}} := \mathcal{P}(\mathcal{V}_t^{\text{sink}}) \cap \mathcal{E}^{\text{cut}}$ and $\mathcal{P}_{t-1}^{\text{sink}} := \mathcal{P}(\mathcal{V}_t^{\text{sink}}) \cap \mathcal{E}^{\text{sink}}$. We have used the subscript $t-1$ here to remind the reader that $\mathcal{P}(\mathcal{V}_t^{\text{sink}})$, which are the *incoming* edges of $\mathcal{V}_t^{\text{sink}}$, are a subset of $\mathcal{E}_{t-1}$. Then, we have

$$\mathcal{P}(\mathcal{V}_t^{\text{sink}}) = \mathcal{P}_{t-1}^{\text{cut}} \cup \mathcal{P}_{t-1}^{\text{sink}}. \tag{52}$$

[19]This is one of the complicating factors that prevents us from recursively applying the Data Processing Inequality at every node, to trace a path backwards from V_{op} to $\mathcal{V}_{\text{ip}}$.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIT.2020.2987806, IEEE Transactions on Information Theory

Since the cut is a zero–M-information cut, we have that for every $E_{t-1} \in \mathscr{P}^{\text{cut}}_{t-1}$,

$$I\big(M; X(E_{t-1}) \,\big|\, X(\mathscr{E}'_{t-1})\big) = 0 \quad \forall\, \mathscr{E}'_{t-1} \subseteq \mathscr{E}_{t-1}. \tag{53}$$

Therefore, by Definition 5 and Proposition 1,

$$I\big(M; X(\mathscr{P}^{\text{cut}}_{t-1}) \,\big|\, X(\mathscr{E}'_{t-1})\big) = 0 \quad \forall\, \mathscr{E}'_{t-1} \subseteq \mathscr{E}_{t-1}. \tag{54}$$

Secondly, $\mathscr{P}^{\text{sink}}_{t-1} = \mathbb{Q}(\mathscr{V}^{\text{sink}}_{t-1}) \cap \mathscr{E}^{\text{sink}}$. This is depicted in Figure 5, and explained in the caption. So,

$$I\big(M; X(\mathscr{P}^{\text{sink}}_{t-1})\big) = I\big(M; X(\mathbb{Q}(\mathscr{V}^{\text{sink}}_{t-1}) \cap \mathscr{E}^{\text{sink}})\big) \tag{55}$$

$$\overset{(a)}{\leq} I\big(M; X(\mathbb{Q}(\mathscr{V}^{\text{sink}}_{t-1}))\big) \overset{(b)}{=} 0 \tag{56}$$

where (a) follows from the fact that considering more random variables can only increase mutual information, and (b) follows from the induction assumption. Finally, consider how $X(\mathbb{Q}(\mathscr{V}^{\text{sink}}_t))$ depends on M:

$$I\big(M; X(\mathbb{Q}(\mathscr{V}^{\text{sink}}_t))\big)$$

$$= I\big(M; f_{\mathscr{V}^{\text{sink}}_t}\big(X(\mathscr{P}^{\text{sink}}_{t-1} \cup \mathscr{P}^{\text{cut}}_{t-1}), W(\mathscr{V}^{\text{sink}}_t)\big)\big) \tag{57}$$

$$\overset{(a)}{\leq} I\big(M; X(\mathscr{P}^{\text{sink}}_{t-1}), X(\mathscr{P}^{\text{cut}}_{t-1}), W(\mathscr{V}^{\text{sink}}_t)\big) \tag{58}$$

$$\overset{(b)}{=} I\big(M; X(\mathscr{P}^{\text{sink}}_{t-1})\big) + I\big(M; X(\mathscr{P}^{\text{cut}}_{t-1}) \,\big|\, X(\mathscr{P}^{\text{sink}}_{t-1})\big) \tag{59}$$

$$+ I\big(M; W(\mathscr{V}^{\text{sink}}_t) \,\big|\, X(\mathscr{P}^{\text{sink}}_{t-1}), X(\mathscr{P}^{\text{cut}}_{t-1})\big)$$

$$\overset{(c)}{=} 0, \tag{60}$$

where once again, (a) and (b) follow from the data processing inequality and the chain rule respectively. In step (c), the first and second terms go to zero by equations (56) and (54) respectively, while the third term is zero since $W(\mathscr{V}^{\text{sink}}_t) \perp\!\!\!\perp \{M, X(\mathscr{E}_{t-1})\}$ and $\mathscr{P}^{\text{sink}}_{t-1} \cup \mathscr{P}^{\text{cut}}_{t-1} \subseteq \mathscr{E}_{t-1}$.

The proof follows from induction on t, so

$$I\big(M; X(\mathbb{Q}(\mathscr{V}^{\text{sink}}_t))\big) = 0 \quad \forall\, t \geq t_0, \tag{61}$$

which in turn implies that

$$I\big(M; X(\mathbb{Q}(V))\big) = 0 \quad \forall\, V \in \mathscr{V}^{\text{sink}}. \tag{62}$$

If there exists an output node whose transmissions depend on M, then there can exist no cut consisting of edges with zero M-information flow, and hence by Lemma 8, there must be a path consisting of edges that carry M-information flow between the input nodes and the output node in question. ■

D. The Separability Property

Finally, we state a property that may be of interest to obtain a deeper understanding of the nature of M-information flow, as given by Definitions 4 and 5.

Proposition 9 (Separability): Let $\mathscr{C}$ be a computational system. Then, at any given point in time t, there exist two sets $\mathscr{R}_t, \mathscr{S}_t \subseteq \mathscr{E}_t$, such that all of the following conditions hold:

1) $\mathscr{R}_t \cup \mathscr{S}_t = \mathscr{E}_t$
2) $\mathscr{R}_t \cap \mathscr{S}_t = \emptyset$
3) Either $\mathscr{R}_t = \emptyset$, or for every $R_t \in \mathscr{R}_t$ there exists a subset $\mathscr{R}'_t \subseteq \mathscr{R}_t \setminus \{R_t\}$ such that
$$I\big(M; X(R_t) \,\big|\, X(\mathscr{R}'_t)\big) > 0. \tag{63}$$
4) Either $\mathscr{S}_t = \emptyset$, or for every $\mathscr{E}'_t \subseteq \mathscr{E}_t$,
$$I\big(M; X(\mathscr{S}_t) \,\big|\, X(\mathscr{E}'_t)\big) = 0. \tag{64}$$

A proof of this proposition can be found in Appendix B.

Proposition 9 shows that at any given point in time t, it is possible to partition $\mathscr{E}_t$ into two sets: $\mathscr{R}_t$, consisting only of edges that have M-information flow, and $\mathscr{S}_t$, comprising edges that have no M-information flow. Furthermore, when considering the M-information flow of edges in $\mathscr{R}_t$, it suffices to condition on the transmissions of edges *within* $\mathscr{R}_t$ to ascertain the presence of M-information flow. Conditioning upon the transmissions of edges in $\mathscr{S}_t$ will not change the mutual information between the message and the transmissions of edges in $\mathscr{R}_t$.

V. Inferring Information Flow

Having discussed the definition and the properties of M-information flow, we now consider how these flows of information might be inferred in a real computational system. We first discuss an observation model that describes which random variables are observed and how they are sampled. Under this model, we show how existing techniques from the literature can be used to identify which edges carry M-information flow. As in previous sections, we restrict our attention to detecting *whether or not* a given edge has M-information flow, relegating quantification of these flows to future work. Quantification is briefly discussed in the form of an example in Section VI-C, and again in Section VII-E.

We then describe an algorithm that recovers all M-information paths between the input nodes and a given output node, by leveraging the knowledge of which edges have M-information flow. We also explain how one might attain a fine-grained characterization of the structure of information flow, by introducing the concept of "derived information". This is useful for understanding which transmissions are "derived" from others, allowing one to find transmissions that are redundant and discover the presence of hidden nodes. Finally, we explain how flows of information about multiple messages can be inferred in our framework.

A. The Observation Model

Before we can describe how information flow and information paths can be identified, we must provide a statistical description of the random variables that are observed. Let $\mathscr{C}$ be a computational system under observation. We then make the following assumptions:

1) Transmissions on all edges, including self-edges, are observed. The random variables that are intrinsically generated at each node are *not* observed, unless they are also transmitted on an edge (which could be a self-edge).
2) Several trials[20] are observed, each of which corresponds to an independent realization of all random variables in

[20]The word "trial" is borrowed from the neuroscience literature, wherein a neuroscientist will often conduct multiple trials in a single experiment. In each trial, a human participant or an animal under study is presented with one of a set of carefully chosen stimuli (corresponding to a realization of the message M in our setting), and neural activity is recorded using some modality. Scientific inferences are then drawn by making use of the activity from all trials.

the model.[21] Every trial uses a realization of M which is independently drawn from a distribution determined by the experimentalist.[22] For every node $V \in \mathcal{V}$, the intrinsically generated random variable $W(V)$ is also assumed to be independently and identically distributed across trials.

3) Observations are made noiselessly, in that the realization of each transmission in every trial is observed as-is, without being further corrupted by random noise of any kind. The implications of noisy measurements will be the subject of future work.

Under these conditions, we discuss statistical tests for information flow that are consistent in the asymptotic limit of infinite trials. It should be noted that these assumptions may be valid to varying degrees in different contexts. This is discussed further in Section VII-A.

B. Detecting Information Flow

Given a sample of all random variables described in the observation model, our next task is to identify which edges have M-information flow. In other words, we need to describe how the conditions given by Definition 4 can be rigorously tested, and how we might assert with some confidence that a certain set of edges has information flow at each point in time.

According to Definition 4, in order to check whether a particular edge E_t carries M-information flow at time t, we need to test whether at least one of several conditional mutual information quantities is strictly positive. The standard statistical approach for solving this problem is to frame it as a set of "hypothesis tests", which in this case is a set of "conditional independence tests". In general, a hypothesis test formalizes the problem of making an informed decision about the value of some functional of a joint distribution, when observing a sample of data from it. A good conditional independence testing procedure will seek to maximize "statistical power", i.e. the probability of *correctly* identifying the presence of conditional dependence, while keeping the probability of an incorrect identification fixed below some "level" α that is picked beforehand. One intuitive way to do this might be to construct an estimator for the appropriate conditional mutual information, and "reject" the "null" hypothesis of conditional independence if the conditional mutual information was sufficiently larger than some threshold, $\epsilon > 0$. This threshold would have to be chosen so that, on average, the probability of falsely rejecting the null hypothesis is at most α. However, there are usually better ways of performing this test, i.e., it is often possible to attain higher power at the same level *without* actually estimating the conditional mutual information.

While it would be impossible to provide a comprehensive list of papers that have researched the problem of conditional independence testing, it has received (and continues to receive) much attention in the statistics, causality, and information theory communities [75]–[80]. In its most general form, conditional independence testing is considered to be a hard problem for continuous random variables [81]. However, if we ignore issues associated with the practical difficulty of estimation (discussed later in Section VII-B), these works provide consistent tests under reasonable assumptions on the joint distribution of the variables involved [77]–[79].

Although we mentioned that there are better ways to test for conditional dependence than to estimate the conditional mutual information, there may be instances when one might want to estimate the conditional mutual information anyway. For instance, in an example that will appear shortly in Section VI-C, we rely on an *estimate* of the conditional mutual information to *quantify* the amount of M-information flowing on a given edge. While our paper has only defined M-information flow in terms of *whether or not* it is present at an edge E_t, it is also extremely useful to know *how much* M-information flow there is. We defer further discussion of this topic until Sections VI-C and VII-E. For now, we note that several papers have considered how to *estimate* mutual information and conditional mutual information, both of which might be essential for an understanding of *quantification* of M-information flow [82]–[85].

For completeness, we now present a description of how we expect information flow will be detected in practice. We assume that we have a sample of the transmissions from all edges of the computational system, at every point in time. If not, appropriate assumptions may need to be made, as discussed later in Section VII-A. At every instant of time t, consider the set of all edges $\mathscr{E}_t$ present in the network. For every edge $E_t \in \mathscr{E}_t$, use the following process to determine whether it has M-information flow:

1) First test whether the mutual information between its transmission and the message is greater than zero, i.e., $I\big(M; X(E_t)\big) > 0$. If so, declare that E_t has M-information flow.
2) If not, test for conditional dependence between its transmission and the message, given each of the other edges E'_t, i.e., test whether $I\big(M; X(E_t) \,\big|\, X(E'_t)\big) > 0$, for each $E'_t \in \mathscr{E}_t \setminus \{\mathscr{E}_t\}$. If any of these tests rejects the null hypothesis, declare that E_t has M-information flow.
3) If not, test for conditional dependence between $X(E_t)$ and M, given subsets of other edges, sequentially conditioning on edges taken pairwise, then in threes, etc. If any of these tests rejects the null, declare that E_t has M-information flow.
4) If none of the above tests rejects the null hypothesis, declare that E_t carries no M-information flow.

Note that we have not discussed the level, α, at which we should reject the null in each of the above tests. In general, since we are performing multiple hypothesis tests simultaneously, some manner of "*correction*" is required to ensure that we do not find, what is effectively, a spurious correlation. This is discussed at length in Section VII-B.

[21] In reality, trials are not independent in neuroscientific experiments. Indeed, neurons are known to "adapt" their responses from trial to trial, often showing suppressed activity when presented the same stimulus multiple times. This, in part, is considered to be evidence of *learning* in neural circuitry. However, for simplicity, we restrict our attention here to computational systems that do not learn or show trial-to-trial adaptation.

[22] A more detailed discussion of this distribution can be found in Section V-F.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIT.2020.2987806, IEEE Transactions on Information Theory

C. Discovering Information Paths

Next, we discuss an algorithm that discovers all M-information paths leading from the input nodes to a given output node, V_{op}, in any computational system. As discussed in Section IV, whenever the transmissions $\mathbb{Q}(V_{\text{op}})$ of the output node depend on the message, Theorem 7 guarantees that at least one M-information path exists.

Algorithm 1, which we propose for recovering all M-information paths, is an adaptation of the well-known Depth-First Search[23] method [74, Sec. 22.3]. It takes as its input a computational system $\mathscr{C}$ in which all edges having M-information flow have been identified, the output node V_{op}, and an empty graph $\mathscr{H}$ that is completely devoid of nodes and edges. The algorithm returns the set of all M-information paths in the form of a directed subgraph $\mathscr{H}$ of the time-unrolled graph $\mathscr{G}$. Starting from $\mathscr{V}_{\text{ip}}$, following *any* path in $\mathscr{H}$ will lead one to V_{op}, provided at least one M-information path exists.

The algorithm works by recursively visiting nodes, starting from the output node V_{op}. It traverses only edges that carry M-information flow, and uses a marking scheme to avoid revisiting nodes. The same marking scheme is also used to designate nodes to which there are M-information paths from $\mathscr{V}_{\text{ip}}$. As the algorithm passes through each node, it marks the node "valid" whenever an M-information path exists between $\mathscr{V}_{\text{ip}}$ and that node. If no such path exists, then the node is marked "invalid". The objective of the algorithm, therefore, reduces to one of finding a path of "valid" nodes from $\mathscr{V}_{\text{ip}}$ to V_{op}. The algorithm's recursive function can be expressed as follows: *A node $V_t \in \mathscr{V}$ is "valid" if and only if there exists a node $U_{t-1} \in \mathscr{V}$ such that U_{t-1} is valid, and the edge (U_{t-1}, V_t) has M-information flow.* This is a recursive expression since checking the validity of a node at time t involves finding valid nodes at time $t-1$. The only nodes that are considered valid by default are the input nodes $\mathscr{V}_{\text{ip}}$.

The algorithm sequentially checks the validity of nodes $V_t \in \mathscr{V}$, starting from the output node V_{op}. The function FINDINFOPATHS, when called on any given node V_t, checks the validity of V_t. This involves checking each of the incoming edges of V_t for M-information flow. If U_{t-1} is a node from which M-information flows to V_t, then the algorithm immediately checks the validity of U_{t-1} by calling the function FINDINFOPATHS again. Eventually, if in this recursive process, we arrive at an input node in $\mathscr{V}_{\text{ip}}$, then that node is marked "valid", and added to the output subgraph $\mathscr{H}$. Once every node U_{t-1} from which M-information flows to V_t has been marked "valid" or "invalid", the validity of V_t can be ascertained. For every "valid" node U_{t-1} from which M-information flows to V_t, the edge (U_{t-1}, V_t) and the node V_t are added to the output subgraph $\mathscr{H}$, and V_t is marked "valid". If there are no such nodes leading to V_t, then V_t is marked "invalid" and does not fall on an M-information path.

This recursive logic yields the set of all M-information paths leading from the input nodes to V_{op}. The two lines at which errors are returned correspond to scenarios that should not occur if the conditions of Theorem 7 hold. In line 12, we visit a non-input node at time $t = 0$. But such a node should never have been reached in the recursion, since we only followed edges that have M-information flow. Its presence, therefore, would contradict the computational system model. In line 4, V_{op} is marked "invalid", implying that there is no path leading to it from the input nodes. Once again, this can only occur if the computational system model is violated, or if the conditions of Theorem 7 (and Property 3) do not hold.

On Computational Complexity: The complexity of Algorithm 1 is exactly that of Depth-first Search, $\mathcal{O}(|\mathscr{V}|+|\mathscr{E}|)$ [74, Sec. 22.3]. To be precise, we consider the computational system to extend until the time of the output node, i.e., we take $T = t_{\text{op}}$. So the complexity of the algorithm is $\mathcal{O}(|\mathscr{V}^*|t_{\text{op}} + |\mathscr{E}^*|t_{\text{op}})$. This is easily verified from the pseudocode listing provided in Algorithm 1: in the worst case, all edges in the system have M-information flow, so all edges and nodes must be traversed by the search. By design of the marking strategy, each node is processed at most once, so we do not need to account for the recursion in any special way. At each node, we must execute lines 7 through 14, and 26 through 28, which take a constant amount of time. Since we have $|\mathscr{V}^*|$ nodes over t_{op} time points, this adds up to $\mathcal{O}(|\mathscr{V}^*|t_{\text{op}})$ steps. We also need to execute the loop in lines 15 through 24, which counts the number of incoming edges at every node. For all nodes combined, this adds up to $\mathcal{O}(|\mathscr{E}^*|t_{\text{op}})$ steps.

If the graph is fully connected as described in Section II, then $|\mathscr{V}^*| = N$ and $|\mathscr{E}^*| = N^2$, so the effective complexity is just $\mathcal{O}(N^2 t_{\text{op}})$. However, if we *know* that the underlying graph is sparse (e.g., because of anatomical priors in neuroscience), then we may have $|\mathscr{E}^*| = \mathcal{O}(N \log N)$, or even $|\mathscr{E}^*| = \mathcal{O}(N)$, bringing down the complexity of the search. It should be noted that in either case, the complexity of identifying *which* edges have M-information flow is potentially exponential in N, as discussed later in Section VII-B. This is much larger than the complexity of tracing out information paths, so *finding edges* with M-information flow is, in fact, the "hard part" of the problem.

D. Derived Information and Redundancy

The framework we develop for information flow allows one to obtain a more fine-grained understanding of information structure in a computational system, especially when compared with classical tools such as correlation and phase synchrony [86], [87]. This allows the experimentalist to better investigate the nature of the computation being performed. A concept that we believe will be extremely useful in this regard is one we call "derived information", which is defined below.

Definition 12 (Derived M-Information): In a computational system $\mathscr{C}$, a transmission $X(Q_t)$ is said to be *derived M-information* of a different transmission $X(P_{t'})$ if M—$X(P_{t'})$—$X(Q_t)$ forms a Markov chain. That is, the following condition must hold:

$$I\big(M; X(Q_t) \,\big|\, X(P_{t'})\big) = 0, \tag{65}$$

implying that

$$H\big(M \,\big|\, X(P_{t'})\big) = H\big(M \,\big|\, X(P_{t'}), X(Q_t)\big). \tag{66}$$

[23] It is also possible to discover all M-information paths using an adaptation of Breadth-First Search [74, Sec. 22.2], but doing so would require some mechanism to prune M-information paths that do not lead to the input nodes $\mathscr{V}_{\text{ip}}$. So we prefer to use Depth-First Search for simplicity of exposition.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIT.2020.2987806, IEEE Transactions on Information Theory

Algorithm 1 Information Path Algorithm: Finds all paths from $\mathscr{V}_{\text{ip}}$ to V_{op}

Initialize an empty graph $\mathscr{H}$ ▷ $\mathscr{H}$ will store valid paths from $\mathscr{V}_{\text{ip}}$ to V_{op}
▷ $\mathscr{H}$ currently contains no nodes or edges
FINDINFOPATHS($\mathscr{C}$, V_{op}, $\mathscr{H}$) ▷ Call a function (defined below) to populate $\mathscr{H}$
if V_{op} is marked "invalid" **then**
    raise Error ▷ No path from $\mathscr{V}_{\text{ip}}$ to V_{op} was found
end if

function FINDINFOPATHS($\mathscr{C}$, V_t, $\mathscr{H}$)
    if $\mathscr{P}(V_t)$ is empty **then** ▷ V_t has no inputs $\Rightarrow t = 0$
        if $V_t \in \mathscr{V}_{\text{ip}}$ **then**
            Mark V_t "valid"
            Add V_t to $\mathscr{H}$
        else ▷ We somehow reached a non-input node at $t = 0$
            raise Error
        end if
    else ▷ V_t has inputs
        for all $(U_{t-1}, V_t) \in \mathscr{P}(V_t)$ **do**
            if (U_{t-1}, V_t) has M-information flow **then**
                if U_{t-1} is unmarked **then**
                    FINDINFOPATHS($\mathscr{C}$, U_{t-1}, $\mathscr{H}$) ▷ This will mark U_{t-1}
                end if
                if U_{t-1} is marked "valid" **then**
                    Mark V_t "valid"
                    Add V_t and (U_{t-1}, V_t) to $\mathscr{H}$
                end if
            end if
        end for
        if V_t is still unmarked **then** ▷ No input of V_t was "valid"
            Mark V_t "invalid"
        end if
    end if
end function

So, $X(Q_t)$ adds no new information about M, when given $X(P_{t'})$. The same definition extends to transmissions on sets of edges. Note that, as far as the definition is concerned, t and t' may be any two arbitrary points in time. However, we will typically consider cases when $t \geq t'$.

One potential use-case scenario for derived information arises in the context of redundant flows. Consider the computational system presented in Figure 4, originally described under Counterexample 3. We see two edges sending the same transmission to the node B_2. This is an example of what we call "redundant transmissions". In general, since we only consider information about M to be relevant, the exact transmissions communicated over two edges at a given point in time may be different. But if they convey the *same information about* M to a given node, then we view them as essentially redundant. Definition 4, when applied to this system, will detect both these edges as having M-information flow, since given $X((C_1, B_2))$, their transmissions depend on M. In the notation of the Separability property mentioned earlier (Proposition 9), both edges (A_1, B_2) as well as (D_1, B_2) will belong in the set $\mathscr{R}_1$.

Derived information provides a general methodology to understand when transmissions on certain edges may be redundant. Naturally, if the transmissions on two edges Q_t and $P_{t'}$ are redundant, then they must be derived M-information of one another. This amounts to checking two more conditional independence relationships, for which consistent tests exist in the limit of infinite trials, as discussed in Section V-B.

In the following section, we shall see another application of derived information; when applied to specific sets, it can in some cases be used to detect the presence of hidden (unobserved) nodes. Later, in Section VI-C, we discuss an example where the notion of derived information helps us make a new kind of inference about the fine structure of information flow, one that would not be possible using tools such as Granger Causality and Directed Information.

E. Hidden Nodes

In Section V-C, we showed how the Information Path Algorithm may fail to discover M-information paths if one of the assumptions of the computational system model or the observation model breaks in some way. Here, we discuss one specific situation in which the observation model may break, i.e., when not all nodes are observed. We call these

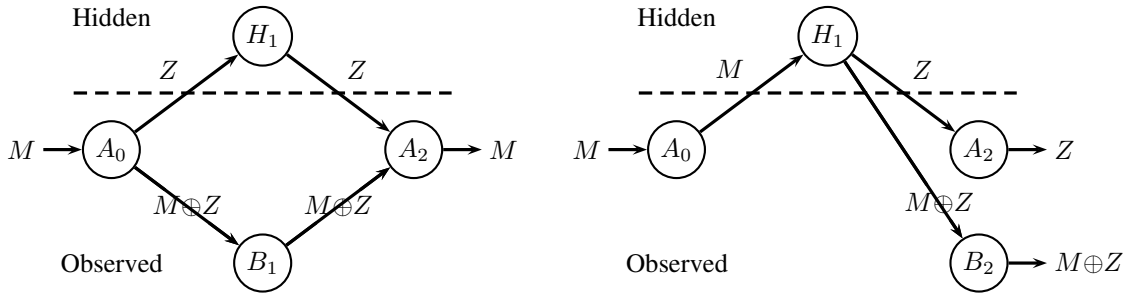


Figure 6: Two simple examples showing how hidden nodes may prevent one from being able to discover M-information paths in a computational system. In both cases shown here, H_1 is a hidden node, and we do not observe its incoming or outgoing transmissions. On the left is an example where a transmission that we might need to condition upon to discover M-information flow passes through the hidden node, and therefore cannot be seen. On the right, the hidden node itself generates the source of randomness Z.

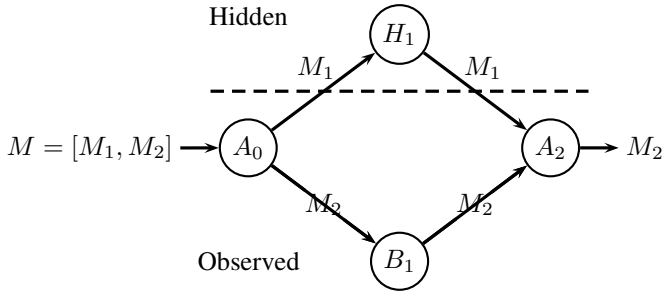


Figure 7: A computational system serving as a counterexample to the converse of Proposition 11. Here, the hidden node H_1 is M-relevant because its outgoing transmission, M_1, is not present in any of the observed transmissions at time $t = 1$. However, since A_2 chooses to ignore M_1 at its output, the Markov chain M—$X(\widetilde{\mathscr{E}}_1)$—$X(\widetilde{\mathscr{E}}_2)$ boils down to M—M_2—M_2, which obviously holds. Thus, at least based on our current definitions, there may be M-relevant hidden nodes in the system even if Global Markovity continues to hold.

unobserved nodes "hidden nodes", and assume that we do not see transmissions on incoming or outgoing edges of these nodes.

Definition 13 (Hidden nodes): Consider a computational system $\mathscr{C} = (\mathscr{G}, X, W, f)$ defined on the time-unrolled graph $\mathscr{G} = (\mathscr{V}, \mathscr{E})$ as before. Suppose that only a subset of nodes in this graph are observed. Specifically, if $\mathscr{V}^*$ was the original set of nodes in $\mathscr{G}^*$, prior to time-unrolling, then we observe only the nodes $\widetilde{\mathscr{V}}^* = \mathscr{V}^* \setminus \mathscr{H}^*$, where $\mathscr{H}^* = \{H^{(0)}, H^{(1)}, \ldots, H^{(K-1)}\}$ is a set of unobserved nodes called *hidden nodes*.

To describe the observed component of the computational system, we define $\widetilde{\mathscr{E}}^* = \widetilde{\mathscr{V}}^* \times \widetilde{\mathscr{V}}^*$, $\widetilde{\mathscr{V}} = \{V_t : V \in \widetilde{\mathscr{V}}^*, t \in \mathscr{T}\}$ and $\widetilde{\mathscr{E}} = \{(A_t, B_{t+1}) : (A, B) \in \widetilde{\mathscr{E}}^*, t \in \mathscr{T}\}$. Also let $\mathscr{H} = \{H_t : H \in \mathscr{H}^*, t \in \mathscr{T}\}$. Finally, we set up the observed component of the computational system as before: $\widetilde{\mathscr{C}} = (\widetilde{\mathscr{G}}, X, W, f)$. Thus, we only observe the transmissions on edges in $\widetilde{\mathscr{E}}$. As usual, we denote the set of all hidden nodes at time t by $\mathscr{H}_t$, and the set of all observed nodes at time t by $\widetilde{\mathscr{E}}_t$.

The presence of hidden nodes of this nature implies that much of the theory we have developed will not apply. Lemma 4 no longer truly holds, in that information about M may persist in the system by passing through the hidden node, even if *no* observed edge has M-information flow. So, naturally, Property 1 also fails to hold. Hence, we are not guaranteed to be able to identify all edges with M-information flow, and discover all M-information paths as before. For example, refer to the cases shown in Figure 6, where we no can longer find M-information paths because of the presence of a hidden node.

Fortunately, at least in some cases, the concept of derived information (Definition 12) provides a simple way to tell whether or not a hidden node exists. Specifically, if at some time t, a hidden node transmits information about M which is unavailable within the system at that time, and which is utilized by some node at the next time instant, then the set of all observed transmissions $X(\widetilde{\mathscr{E}}_t)$ will *not* be derived M-information of the set of all transmissions at time $t-1$. In other words, the Global Markovity condition (Corollary 5) on the observed graph, M—$X(\widetilde{\mathscr{E}}_{t-1})$—$X(\widetilde{\mathscr{E}}_t)$, will break. Unfortunately, the notion of "utilization" is difficult to express mathematically, without resorting to the use of ideas from causality that are based on intervention. The result we prove, therefore, is a simpler sufficiency argument, which guarantees the presence of a hidden node if the aforementioned Markov condition is observed to break. This result is proved in Proposition 11, but first, we define some adjectives.

Definition 14 (M-relevant hidden node): A hidden node H_t is said to be *M-relevant* if $\mathbb{Q}(H_t)$ carries M-information flow in $\mathscr{G}$. Similarly, a subset of hidden nodes $\mathscr{H}'_t \subseteq \mathscr{H}_t$ is said to be M-relevant if $\mathbb{Q}(\mathscr{H}'_t)$ carries M-information flow in $\mathscr{G}$.

Definition 15 (M-derived hidden node): A hidden node H_t is said to be *M-derived* if the Markov chain M—$X(\widetilde{\mathscr{E}}_t)$—$X(\mathbb{Q}(H_t))$ holds. Similarly, a subset of hidden nodes $\mathscr{H}'_t \subseteq \mathscr{H}_t$ is said to be M-derived if the Markov chain M—$X(\widetilde{\mathscr{E}}_t)$—$X(\mathbb{Q}(\mathscr{H}'_t))$ holds.

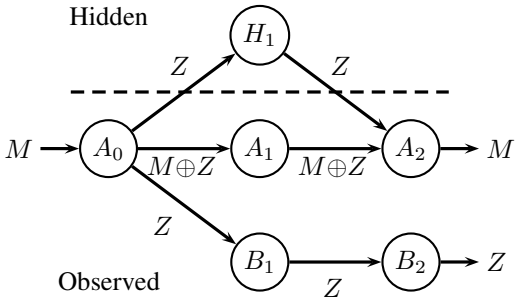
Lemma 10: If a subset of hidden nodes is *not* M-derived, then it *is* M-relevant.[24]

Proposition 11: In a computational system $\mathscr{C}$ with hidden nodes, if Global Markovity on the observed graph, $\widetilde{\mathscr{G}}$, fails to hold from time t to $t+1$, i.e. if $I\big(M; X(\widetilde{\mathscr{E}}_{t+1}) \,\big|\, X(\widetilde{\mathscr{E}}_t)\big) > 0$, then the hidden nodes $\mathscr{H}_t$ at time t are *not* M-derived.

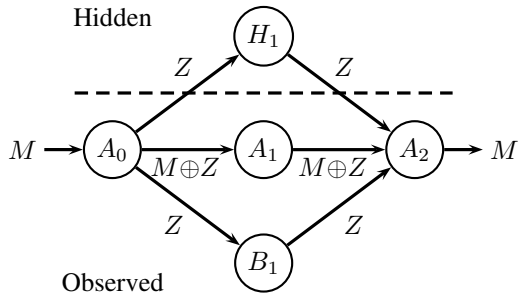
Proofs of Lemma 10 and Proposition 11 are straightforward, and are provided in Appendix D. As a direct consequence of

[24]If this lemma appears to be somewhat strong, it is only because of the nomenclature "M-derived". For our purposes, a hidden node whose transmissions are independent of the message is also M-derived, since it satisfies the aforementioned Markov condition.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIT.2020.2987806, IEEE Transactions on Information Theory



(a) Here, although Global Markovity holds, one could argue that testing for Local Markovity at each node (or at various subsets of nodes) could help uncover the presence of a hidden node.



(b) In this case, the hidden node breaks neither Global nor Local Markovity. However, the *function* computed by A_2 makes *use* of only the hidden node's transmission. As a result, the hidden node has a causal effect on the output of the system, since destroying the outgoing edge of the hidden node would change the output. Such a hidden node is likely undetectable using only observational methods.

Figure 8: Examples of computational systems with an M-derived hidden node. In both of these systems, the hidden node's transmission at time $t = 1$ has an effect on the output at A_2. However, Global Markovity continues to hold from $t = 1$ to $t = 2$, because the observed transmissions, $M \oplus Z$ and Z, contain all information necessary to explain the output, M.

these two results, if Global Markovity fails to hold on the observed nodes from time t to $t + 1$, then $\mathcal{H}_t$ is M-relevant. By Proposition 1, this simply means that there exists at least one M-relevant hidden node at time t.

Although Proposition 11 appears to provide a straightforward mechanism to test whether or not hidden nodes exist, it does not always work. If a hidden node's transmissions have no M-information flow, then the node will not be detected. But in this case, it could be argued that such a hidden node does not change whether information paths can be identified, and so can be subsumed by one or more of the intrinsic random variables $W(\cdot)$. Such a hidden node is, therefore, classified by Definition 14 as *not* M-relevant.

However, to make matters worse, the converse of Proposition 11 does not hold. In particular, there may exist an M-*relevant* hidden node at time t, whose transmission is *ignored* by the node that received it, so that the Markov chain M—$X(\widetilde{\mathcal{E}}_t)$—$X(\widetilde{\mathcal{E}}_{t+1})$ continues to hold (see Figure 7). Such a hidden node may *still* be considered largely innocuous.

The most serious case of a hidden node going undetected is one that contains an M-*derived* hidden node, whose transmission *is used* by the receiving node while performing its computation; however the hidden node's transmission is "masked" by a redundant transmission from an observed node (see Figure 8). In this case, Global Markovity on $\widetilde{\mathcal{G}}$ will *not* break, yet the hidden node's transmission may be instrumental in producing a certain output distribution. In some instances, such hidden nodes can be detected by checking for Local Markovity (Proposition 6; see Figure 8a). However, there are still cases where if we were somehow able to intervene and delete the transmission of the hidden node, then the computational system's output may not remain the same, despite the existence of a redundant transmission from an observed node (see Figure 8b). Indeed, the presence of redundancy in such a scenario does not guarantee that the computational system will actually leverage it.

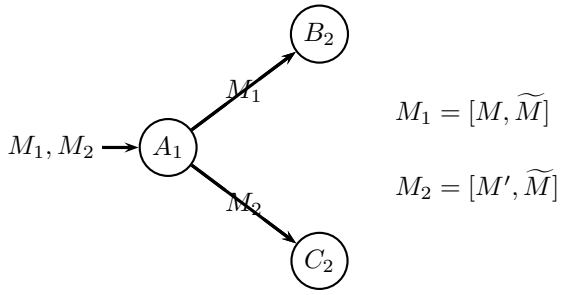


Figure 9: A simple example demonstrating the importance of having independent messages (or sub-messages) when exploring the flows of multiple messages in a computational system. As M_1 and M_2 both redundantly contain information about $\widetilde{M}$, both edges shown here have M_1- as well as M_2-information flow. Thus, we are unable to detect the fact that M_1 and M_2 take different paths in the system, because of our choice of stimuli.

F. On Multiple Messages and the Distribution of the Message

Just as we can infer information flow and information paths for a single message, we can examine the flows of multiple messages in the same computational system. Consider a case where we wish to understand the information flows of two messages, M_1 and M_2. An neuroscientific example of this might be information flow about two independent components of a visual stimulus, e.g., shape and color (such as in [2]). If $M_1 \perp\!\!\!\perp M_2$, then we could separately identify edges and paths that have M_1-information flow and M_2-information flow, by applying the theory and algorithm as-is for each message individually.

However, if the two messages are *dependent* on one another, one could end up conflating their information flows, based on how they depend on each other, and how the computational system's transmissions carry their joint information. As a simple example, consider the system shown in Figure 9, where $M_1 = [M, \widetilde{M}]$ and $M_2 = [M', \widetilde{M}]$, with $M, M', \widetilde{M} \sim$ i.i.d. Ber$(1/2)$. Clearly, M_1 and M_2 both share some redundant information in $\widetilde{M}$, and $I(M_1; M_2) = 1$ bit. Thus, we will see M_1-information flow as well as M_2-information flow on

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIT.2020.2987806, IEEE Transactions on Information Theory

both edges, since the transmission of each edge E satisfies $I\big(M_i; X(E)\big) > 0$ for $i \in \{1, 2\}$.

Consider what this means for the aforementioned example of shape and color of a visual stimulus. If a neuroscientist expects that the information paths corresponding to shape and color in the brain are different from each other, what is the best way to design stimuli so as to bring out this difference? Suppose they decided to present a total of four different stimuli, $M \in \{0, 1, 2, 3\}$, with two different shapes and two different colors. Let M_1 be the first bit of the binary representation of M, denoting shape, and M_2 be the second bit, denoting color. Now if the neuroscientist chose to present stimuli with a uniform distribution over M, i.e., if each shape-color combination was shown for one-quarter of all trials, then M_1 and M_2 would be independent of each other, and their individual flows could be tracked separately. However, if the neuroscientist chose to present the four possible stimuli with probabilities $\{1/2, 1/4, 1/8, 1/8\}$ respectively, then M_1 and M_2 are no longer independent of each other, and it may become hard to separate their individual flows as in the example in Figure 9.

These examples suggest that, when trying to understand the flows of different messages in a computational system, it helps if they are independent of one another. So from the perspective of experiment design in a neuroscientific context, it is often more sensible to design stimuli so that the two messages of interest are independent of one another. Even when considering a single message that takes one of several values, it becomes important to appropriately choose a distribution over these values to ensure that any sub-messages that are of interest remain independent of one another. This would allow the experimentalist to better understand how "independent dimensions" of the stimulus are processed in the brain.

However, there are also situations where the experimental paradigm necessitates a statistical distribution of stimuli that makes two sub-messages of interest dependent on one another. For instance, the Posner experimental paradigm for attention [88] only works when the proportion of "valid" trials (a certain *type* of trial specific to this paradigm) is roughly 70%. Similarly, during data preprocessing, it is common to discard trials that are excessively noisy, based on some predetermined metric: this process could skew the distribution of the message, even if the original distribution was uniform. If it is still of interest to understand the individual flows of sub-messages in this case, then a possible solution might then be to sub-select experimental trials in such a way as to keep the two sub-messages independent of one another.

VI. Canonical Computational Examples

In this section, we provide a few canonical examples for computational systems from various contexts. In each case, we discuss what the message M is, and identify which edges carry M-information flow. We also explain how the path recovered by the information path algorithm might be the intuitive choice in each example.

A. The Butterfly Network from Network Coding

For our first example, we cover the butterfly network from the network coding literature [7, Fig. 7b], reproduced here in Figure 10. In this system, we want to communicate two independent bits, $M_1, M_2 \sim$ i.i.d. Ber$(1/2)$, from C_0 to two output nodes, A_4 and B_4. In the network coding context, the butterfly network is the canonical example illustrating that "coding" is necessary to achieve optimal communication: when each edge is restricted to have a capacity of 1 bit, it is not possible to send both M_1 and M_2 simultaneously to A_4 and B_4 using *routing* alone, since we can send only one of M_1 or M_2 on the middle branch (C_2, C_3). We must use *coding*, i.e., we must compute a function of M_1 and M_2, in order to communicate both message bits to A_4 and B_4.

We examine the individual information flows of M_1 and M_2 in the maximal-rate setting where the middle branch carries $M_1 \oplus M_2$. Edges along which information about M_1 flows are colored in blue, while edges along which information about M_2 flows are colored in orange. The reader may identify these using Definition 4 and the transmission on each edge shown in Figure 10.

An important feature to observe is that when C_2 mixes information by computing the XOR of M_1 and M_2, we see information about M_1 spontaneously beginning to flow on (B_2, B_3) and similarly, information about M_2 beginning to flow on (A_2, A_3). This is expected, since M_2 is relevant for decoding M_1 at this stage, and indeed, it is exactly this idea which is used to decode M_1 at B_4. All of this is true, despite the fact that $M_1 \oplus M_2$ is independent of M_1 and M_2 individually. This is once again, a prime example of synergy in action.

Applying the information path algorithm (Algorithm 1) for the message M_1 at A_4 will reveal two paths: the "upper path" $(C_0, A_1, A_2, A_3, A_4)$, and the "middle path" $(C_0, A_1, C_2, C_3, A_4)$. However, applying the information path algorithm for the other message M_2 at the same output node A_4 reveals that M_2 exclusively uses the "middle path", $(C_0, B_1, C_2, C_3, A_4)$, to arrive at A_4 from the input nodes.

B. The Fast Fourier Transform

The Fast Fourier Transform (FFT) is a well-known computational network that provides an intuitive setting for examining information flow. In general, the N-point FFT is an implementation of the N-point Discrete Fourier Transform (DFT), given by

$$\widetilde{Y}_k = \sum_{i=0}^{N-1} Y_i e^{-j\frac{2\pi k}{N} i}, \quad k \in \{0, 1, \ldots, N-1\} \tag{67}$$

where j is the imaginary unit. The DFT is a basis transformation of a discrete-time signal Y, which is usually assumed to be periodic with period N. The N-point DFT represents such a signal in the complex-exponential Fourier basis, yielding the Fourier coefficients $\widetilde{Y}$. We consider a simple 4-point DFT, i.e. $N = 4$. The FFT implements this transform using the computational system shown in Figure 11. We refer the reader to [62, Ch. 9] for details. For notational convenience, we have set $\omega = e^{-j\frac{2\pi}{N}} = e^{-j\frac{2\pi}{4}}$.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIT.2020.2987806, IEEE Transactions on Information Theory

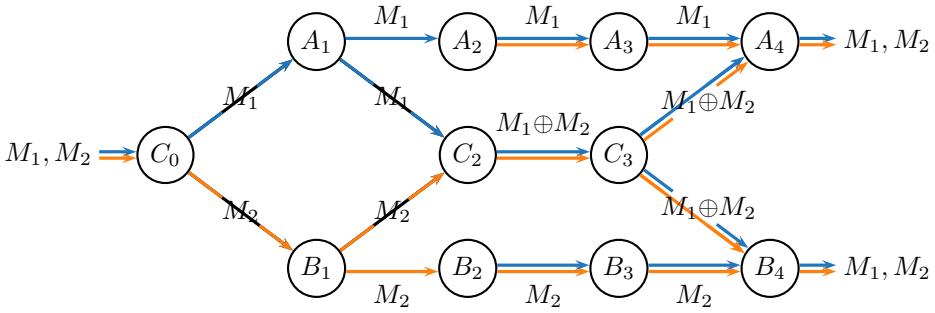


Figure 10: A depiction of the butterfly network discussed in Section VI-A. There are two messages, M_1 and M_2, each with its own information flow. All edges with M_1-information flow are shown in blue and all edges with M_2-information flow are shown in orange. After time $t = 2$, all edges shown have both M_1- and M_2-information flow. Once the system computes $M_1 \oplus M_2$, edges transmitting M_1 have information flow about both M_1 and M_2, since M_2 can now be decoded from $M_1 \oplus M_2$ and M_1. Furthermore, observe the M_1- and M_2-information paths in this system. In particular, there are two possible M_1-information paths to A_4, but only one possible M_2-information path, which flows through the middle link. The same applies to the M_1-information path to B_4. This may suggest the importance of the middle link in enabling this computation.

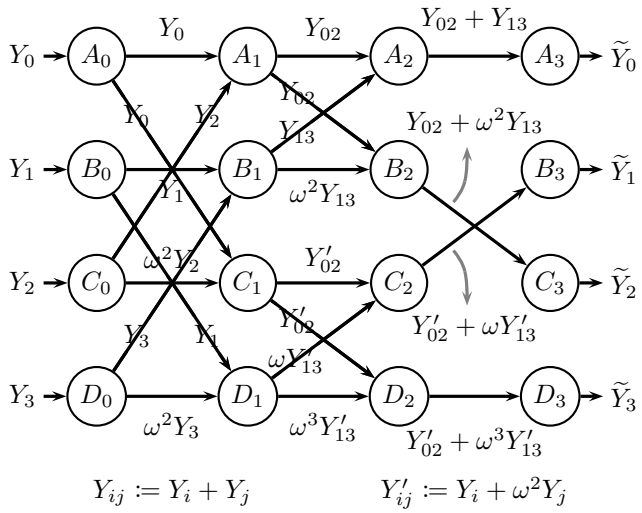


Figure 11: The computational system of the 4-point Fast Fourier Transform. For brevity, we have set $\omega \coloneqq e^{-j\frac{2\pi}{4}}$.

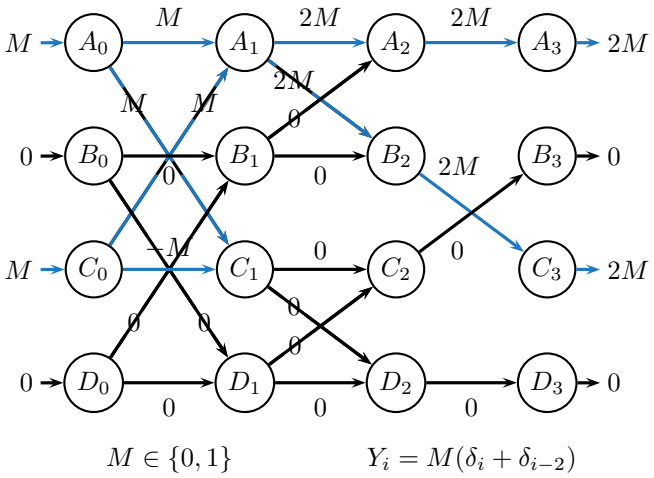


Figure 12: An example of information flow in the 4-point FFT, when the message determines which of two signals is supplied to the system: $Y = [0, 0, 0, 0]$ or $Y = [1, 0, 1, 0]$. Observe that, since M is encoded in the even part of Y, only the "even component" of the FFT network is active. Furthermore, only the DC component, $\widetilde{Y}_0$ and the first harmonic, $\widetilde{Y}_2$ are active, as we would expect based on the two input signals.

We use this example to demonstrate how the definition of the message is important in determining information flow. First, suppose the message is one of two signals: $Y = [0, 0, 0, 0]$, or $Y = [1, 0, 1, 0]$. This can be written as $M \in \{0, 1\}$ and $Y_i = M(\delta_i + \delta_{i-2})$, where $\delta_i = \mathbb{I}\{i = 0\}$ is the Kronecker Delta function, and we assume $M \sim \text{Ber}(1/2)$. The full computational system, along with the random variables computed on all edges, is shown in Figure 12. The edges that have M-information flow are highlighted in blue. Since M is encoded into the *even* part of Y (observe that $Y_i = Y_{-i} \ \forall \ M$), we notice that only the "even component" of the FFT system (corresponding to the 2-point FFT on the even indices of Y) is active [62, Sec. 9.3]. Furthermore, only $\widetilde{Y}_0$, the DC component, and $\widetilde{Y}_2$, the first harmonic, show variation with M at the output, as we would expect based on what differs between the two input signals.

As a second example, consider the case shown in Figure 13. Here, the message is again one of two signals: $Y = [1, 1, 1, 1]$, or $Y = [1, 1/\omega, 1/\omega^2, 1/\omega^3]$. These signals can be jointly expressed in terms of the binary message random variable $M \sim \text{Ber}(1/2)$ as $Y_i = 1/\omega^{iM}$. The two signals are flat in their magnitude spectra and differ only in their phase, creating δ-functions in the Fourier domain that are frequency-shifted with respect to one another: $\widetilde{Y}_k = \delta_{k-M}$. Once again, the edges in the network that carry M-information flow are demarcated in blue. Refer Appendix F-A for a derivation of the values of the transmissions in the computational system.

These two examples make it clear that, based on how the message is defined, the M-information paths in the system can be very different. Indeed, if the message were as general as possible, by placing a probability distribution over all possible values of Y in $\mathbb{R}^4$, we know that *all* edges in the computational system would have M-information flow. However, selectively restricting M to just a few signals helps reveal some kind of structure within the FFT network.

Another feature that can be observed in these examples is how the output of the computational system can be a *function* of the message. Although only very simple functions of the message have been shown at the outputs here, the FFT demonstrates that, in principle, more complex functions of the message may also be generated.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIT.2020.2987806, IEEE Transactions on Information Theory

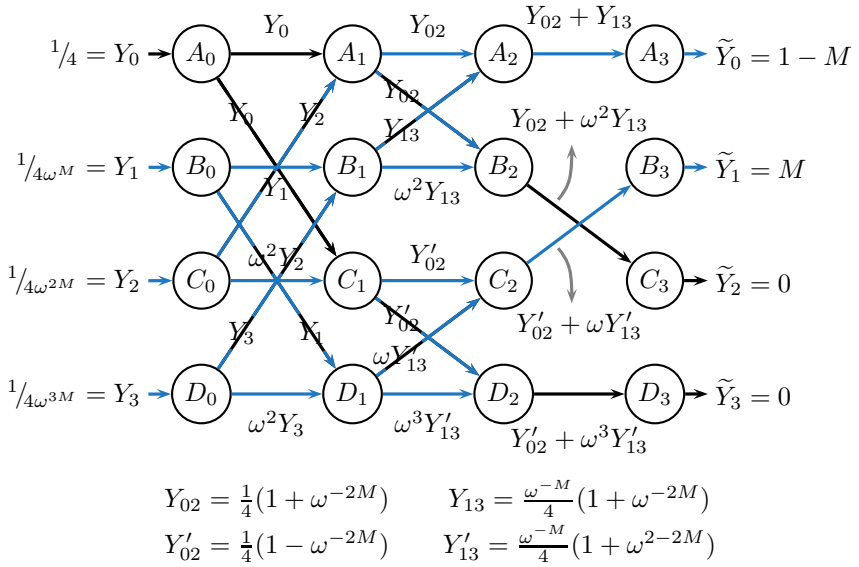


Figure 13: Another example of information flow in the 4-point FFT, when the message determines which of two signals is supplied to the system: $Y = [1, 1, 1, 1]$ or $Y = [1, 1/\omega, 1/\omega^2, 1/\omega^3]$. The M-information paths are different from those in Figure 12, showing how the choice of the message can have a strong impact on the flows within the same computational system.

C. The Schalkwijk and Kailath Scheme

The Schalkwijk and Kailath scheme [89] is an efficient strategy for communicating a message in the presence of a noisy feedforward channel and a noiseless feedback channel. We have previously used this scheme as a counterexample [58], to show that comparing Granger causal influences in forward and backward directions can lead to erroneous inferences on the direction in which the message is being sent in this feedback system. We first provide a brief overview of the scheme, then recapitulate our previous result, and finally demonstrate what the information flow framework developed in this paper has to offer in the case of this example.

Consider the communication system depicted in Figure 14, which shows the schematic of a simplified version of the Schalkwijk and Kailath scheme. For convenience, let us denote the transmitter, A, and receiver, B, by Alice and Bob respectively. Alice is attempting to communicate a message M to Bob over an additive Gaussian channel, but in the presence of noiseless feedback. Alice starts by transmitting the message $Y_1 = M$ to Bob, over the noisy feedforward channel. Bob receives a corrupted version of M, given by $\widetilde{Y}_1 = Y_1 + Z_1$, and computes an estimate $\widehat{M}_1$. He sends this estimate back to Alice over the noiseless feedback channel. In the iterations that follow, Alice computes the error in Bob's most recent estimate, $Y_i = M - \widehat{M}_{i-1}$, and sends this to Bob over the noisy feedforward channel. Meanwhile, Bob updates his estimate based on Alice's noisy transmissions $\widetilde{Y}_i = Y_i + Z_i$, using the following rule:

$$\widehat{M}_i = \widehat{M}_{i-1} + \frac{1}{i}\widetilde{Y}_i \tag{68}$$

It can be shown that this rule implies

$$\widehat{M}_i = M + \frac{1}{i}\sum_{j=1}^{i} Z_j \tag{69}$$

Thus, this strategy ensures that Bob's estimate $\widehat{M}_i$ converges to M in mean squared sense [58].

Intuitively, one might expect that, since the message M is being transmitted in the forward direction, the Granger causal influence from Alice to Bob is greater than that from Bob to Alice. However, our earlier result [58] showed that, in fact, the opposite is true. In other words, even though the message is being communicated from Alice to Bob, the Granger causal influence from Bob to Alice is greater; in fact, the Granger causal index from Bob to Alice is *infinite*. The reason for this is that, while Alice's past transmissions do not perfectly predict Bob's transmissions (due to the presence of noise in the feedforward link), Bob's past transmissions *perfectly* predict Alice's transmissions (since the latter are a simple function of the former). Therefore, the Granger causal index from Alice to Bob, which measures the relative predictive gain of including Alice's past transmissions in the autoregression for Bob's transmissions, remains finite; while the Granger causal index from Bob to Alice becomes infinite.

Our earlier paper on this subject [58] concluded that the direction of greater Granger causal influence could be opposite to the "direction of information flow" in the Schalkwijk and Kailath scheme. There, "information flow" was being used purely in an intuitive sense, to mean the direction in which the message was being communicated in that system. The intent of our previous paper was to explain that it is not always possible to interpret a larger Granger causal influence in a certain direction to mean that a specific message is being communicated in that direction. In contrast, this paper presents a refined theoretical framework that defines information flow about a message M for a specific *edge* in a computational system. Now, we no longer speak of *one specific direction* in which information flows; rather, we describe *which edges* carry information about the message in their transmissions *at each point in time*. This leads to a more nuanced understanding of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIT.2020.2987806, IEEE Transactions on Information Theory

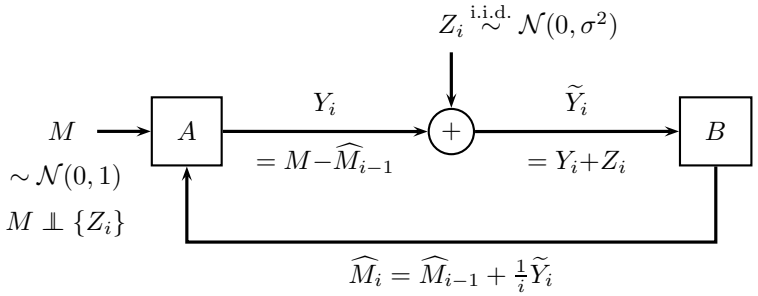


Figure 14: A communication system depicting the Schalkwijk and Kailath scheme. Alice, represented by node A, communicates a message M to Bob, represented by node B, in the presence of a noisy feedforward channel and a noiseless feedback channel. In the i^{th} iteration, Alice transmits the error in Bob's most recent estimate of the message, Y_i, but her transmission is corrupted by the noise Z_i. Bob updates and transmits his estimate, $\widehat{M}_i$, which reach Alice noiselessly.

information flow in the Schalkwijk-Kailath setting.

Before we can analyze the M-information flows in the Schalkwijk-Kailath scheme, we need to fit the scheme within the computational system framework. Figure 15 shows the time-unrolled computational system corresponding to two feedforward and feedback iterations of the simplified Schalkwijk-Kailath scheme described before. In order to translate the communication system into our computational system model while remaining consistent with our earlier work [58], we have merged the process of noise addition with the receiver, i.e., Bob. This exposes the edges with Alice's and Bob's *transmissions*, making them observable, as was assumed in our previous paper [58]. This is also consistent with what *would have been* observable if A and B were neurons (or neural populations) whose outputs a neuroscientist were to measure.[25] Note that one full iteration of the Schalkwijk-Kailath scheme takes two time steps in this model, so the iteration index i advances once for every two time steps t. Also, note that merging noise-addition with the receiver does *not* make $\widetilde{Y}$ or Z "hidden nodes", since the function computed at B_t can be defined purely in terms of its inputs, $(Y_i, \widehat{M}_{i-1})$, and its intrinsic random variable, $W(B_t)$ (which absorbs Z_i), as follows:

$$f_{B_{2i-1}}\big(Y_i, \widehat{M}_{i-1}, W(B_{2i-1})\big) = \widehat{M}_{i-1} + \frac{1}{i}\big(Y_i + W(B_{2i-1})\big) \tag{70}$$

where $W(B_{2i-1}) = Z_i$ takes the role of the noise in the communication system. Also, to understand the time index for node B, note that in the first step of iteration i, Alice transmits to Bob, i.e., node A_{2i-2} transmits to B_{2i-1} (see Figure 15).

Now, we first show that all edges depicted in blue in Figure 15 carry M-information flow, based on Definition 4. Specifically, both Alice's feedforward transmissions *and* Bob's feedback transmissions have M-information flow. This should not be surprising for the following intuitive reasons: Alice's transmissions convey information about M which Bob uses to improve his estimate; meanwhile, Bob's transmissions are estimates of M, and therefore must depend on M.

[25]From a wireless communication system perspective, as well, it is more reasonable to assume noise to be a part of the receiver's node, since the additive noise in a signal is usually considered to be the result of thermal noise in the receiver's circuitry.

In fact, we can take this intuitive argument further: suppose we were to *quantify* M-information flow by using the following natural extension of our definition,

$$\mathcal{F}_M(E_t) \coloneqq \max_{\mathscr{C}_t' \subseteq \mathscr{C}_t} I\big(M; X(E_t) \,\big|\, X(\mathscr{C}_t')\big). \tag{71}$$

Noting that Definition 4 only specified *whether or not* a given edge E_t had information flow, all that we have now done is to take the maximum over the subsets of edges used to discover M-information flow in that definition. This quantification is fully consistent with our definition of M-information flow, since it goes to zero if and only if the M-information flow on an edge goes to zero. Now, using this quantitative notion of information flow, we can ask how the M-information flow on a given link—feedforward or feedback—varies with time. In particular, it should be intuitively clear that the M-information content in Bob's transmissions, i.e. $\widehat{M}_i$, *increases* over time as his estimate improves. This is depicted as an increase in the thickness of the edges carrying Bob's transmissions with time.

On the other hand, the information content in Alice's transmissions *decreases* with time. To understand why this is true, first note that $I(M; Y_i) = 0$ for $i > 1$, since Y_i carries only information about the noise in $\widehat{M}_{i-1}$ (after the first iteration), which is independent of M, as seen in Equation (69). However, since Alice's transmissions represent the *noise* in Bob's estimates of M, they depend on the message when *conditioned* on Bob's estimates (this is similar to how Z carries information about M when conditioned on $M \oplus Z$ in Counterexample 1). So the quantified M-information flow of Alice's transmissions will be given by:

$$\begin{aligned}
&I(M; Y_i \,|\, \widehat{M}_{i-1}) \\
&\quad = I(M; M - \widehat{M}_{i-1} \,|\, \widehat{M}_{i-1}) && (72)\\
&\quad \overset{(a)}{=} H(M \,|\, \widehat{M}_{i-1}) + H(M \,|\, M - \widehat{M}_{i-1}, \widehat{M}_{i-1}) && (73)\\
&\quad = H(M \,|\, \widehat{M}_{i-1}) && (74)\\
&\quad = H(M) - I(M; \widehat{M}_{i-1}) && (75)
\end{aligned}$$

where in (a), the second term goes to zero because M is a constant when given $\widehat{M}_{i-1}$ and $M - \widehat{M}_{i-1}$. During the initial iterations, when Bob's estimate is poor, we must have that $I(M; \widehat{M}_{i-1})$ is very small (as we might expect if the noise is large, for instance). Hence, for the first few iterations, the quantified M-information flow of Alice's transmissions will

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIT.2020.2987806, IEEE Transactions on Information Theory

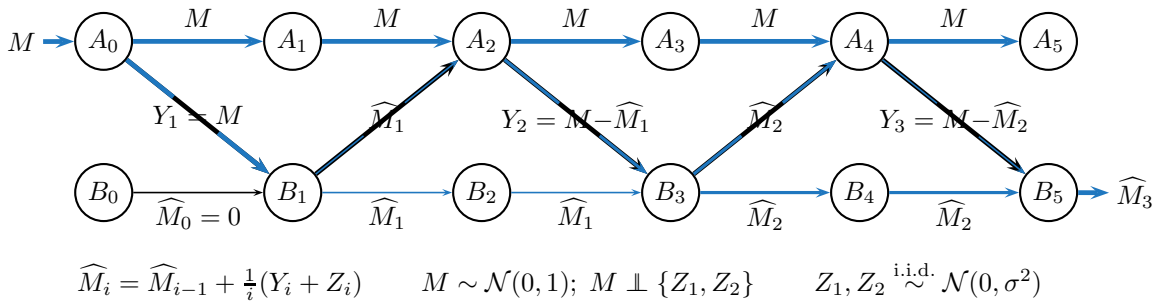


Figure 15: A computational system describing the first few iterations of the Schalkwijk and Kailath scheme. Almost every edge shown here has M-information flow. However, the *quantity* of M-information flow (shown using line thickness) reveals the asymmetry between Alice and Bob: Alice has the message to begin with, and her transmissions have a larger volume of M-information flow. In contrast, Bob's initial transmissions are poor estimates and have small volumes of M-information flow, but they get better over a few iterations, and eventually come close to the true message. Furthermore, we also reveal an asymmetry between Alice and Bob using the concept of derived information: each of Bob's transmissions is M-derived from Alice's previous transmissions, whereas Alice's transmissions are *not* M-derived from Bob's previous transmissions. Both these facts point towards the idea that Alice is slowly sending information about M to Bob.

be close to $H(M)$, from Equation (75). However, as Bob's estimate improves, $I(M; \widehat{M}_{i-1})$ becomes closer to $H(M)$, and therefore $I(M; Y_i \,|\, \widehat{M}_{i-1})$ becomes close to zero. Thus, the quantified M-information flow of Alice's transmissions decreases over time. Correspondingly, this is depicted using edges whose thickness decreases over time in Figure 15. Quantifying the M-information flows of the feedforward and feedback links thus reveals an asymmetry between Alice and Bob that strongly suggests that the message is being transmitted from Alice to Bob.

We can also get a more nuanced understanding of information flow in this system by asking whether Bob's transmissions are *derived* from Alice's, or vice versa. First, consider whether Bob's transmissions are derived M-information of Alice's previous transmissions: this can be expressed in terms of the Markov chain M—$[M,\ M - \widehat{M}_1]$—$\widehat{M}_2$. Observe that this Markov chain holds trivially:

$$I(M; \widehat{M}_2 \,|\, M,\, M - \widehat{M}_1) = 0. \tag{76}$$

However, if we consider whether Alice's transmissions are derived M-information of Bob's past transmissions, it can be shown that M—$[\widehat{M}_1,\ \widehat{M}_2]$—$(M - \widehat{M}_2)$ is not a valid Markov chain (see Appendix F-B for a detailed derivation). Hence, we see that Bob's transmissions are derived M-information of all of Alice's past transmissions, however, Alice's transmissions are *not* derived M-information of all of Bob's past transmissions. In conjunction with the fact that the volume of M-information flow in Alice's transmissions slowly decreases from $H(M)$ with time, while the volume of M-information flow in Bob's transmissions slowly increases to $H(M)$ with time, this suggests that Alice has some information about the message M that Bob slowly receives from Alice.

This example shows how a measure that quantifies information flow, along with derived information, can be used to understand some finer computational structure present within the computational system. In general, however, care needs to be exercised in applying derived M-information: one must choose what Markov condition to check in a principled manner. In the specific case of the Schalkwijk-Kailath example, we had the advantage of being in a two-node setting, where the derived information expressions we examined had clear interpretations. It may be that analyzing information flow first, to understand which variables transmit information about M to one another, can help guide the choice of variables to examine when applying derived M-information.

D. A Message Defined at the Output of a System

We now describe an example where the message is defined at the *output* of a computational system, instead of at the input. Although Definition 3c defines the message to be a random variable available at the input nodes, it is also possible to define the message at the output of the computational system. In this scenario, the input nodes are no longer well-defined as per Definition 3c. Instead, we would define *output nodes* in the same manner.[26]

Consider the computational system shown in Figure 16. The system on the right executes the function depicted by the boolean circuit shown on the left. $Y \in \{0, 1\}$ is an external parameter, which is taken to be a fixed constant. When $Y = 1$, the AND gate at the top is activated while the AND gate at the bottom is deactivated, so the message depends only on Z_1. In this case, only the edges shown in blue have M-information flow. On the other hand, when $Y = 0$, the opposite happens, and the message M depends only on Z_2. Now, only edges shown in orange have M-information flow. If Y was not a deterministic external parameter, but a random variable itself, then all edges shown in the figure would have M-information flow, since M would depend on all their values.

So, we see that when the message is defined at the output, the "origin" of the message may be from within the computation system itself, in the form of one or more intrinsically generated random variables: here, either $Z_1 = W(A_0)$ or $Z_2 = W(C_0)$. The notion of information flow and information

[26]Note, however, that the corresponding "opposite" of Theorem 7 does not hold in this case. That is, it is *not true* that if at some previous time instant, an "*input*" node's outgoing transmissions depend on the message, then there exists an information path connecting that input node to the aforementioned output nodes. The reason this fails is that there could be a "source" node at an even earlier time instant, which provides information about M to *both* the input node under consideration, and the output nodes, via two separate, diverging paths. Therefore, there may be no path from said input node to the output nodes.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIT.2020.2987806, IEEE Transactions on Information Theory

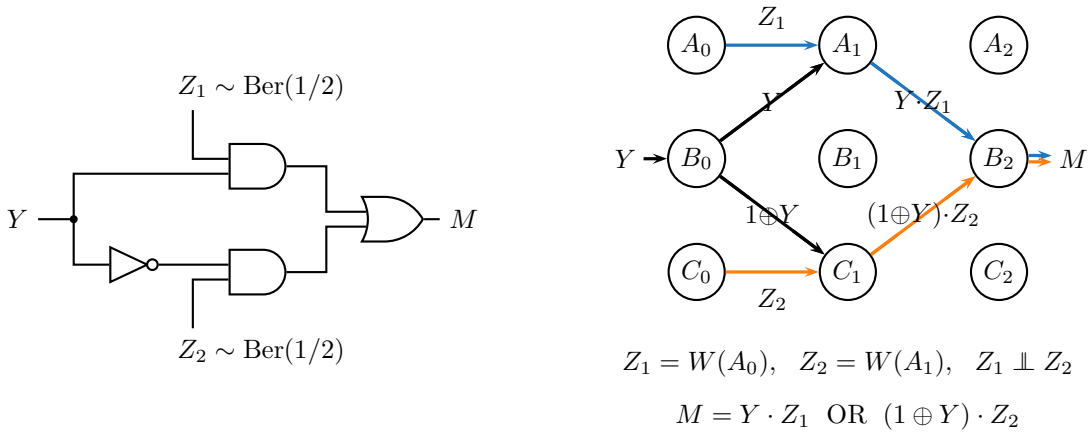


Figure 16: A boolean circuit demonstrating a message defined at the output of the computational system. Note that "$\oplus$" refers to bitwise-XOR, "OR" refers to bitwise-OR, and "$\cdot$" refers to bitwise-AND. We see that information paths may lead from an internal node, that generates an intrinsic random variable, to the output node. Furthermore, this path may change with the "external parameters" of the system.

paths can thus help us identify where the message originates within the computational system.

Furthermore, just as information paths can change depending upon how the message is defined (as in Section VI-B), information paths may also change depending on external parameters: inputs such as Y that are fed into the computational system, which are not part of the message. These inputs essentially shape the nature of the computation being performed, and so, naturally, they can affect information paths.

VII. Discussion

This paper presented a theoretical framework for defining and studying information flow about a specific message in a computational system. The core contribution of our paper was a definition for information flow that is concretely grounded in the *computational task* and intimately tied to a *specific message*. This relied on another important contribution: the development of an underlying computational model, which enables the interpretation of statistical analyses. After providing a clearly-defined model for a computational system, we presented several candidate definitions for information flow along with counterexamples and showed that our definition, which is based on positivity of a conditional mutual information expression, satisfies several intuitive properties, whereas other candidate definitions do not. We then examined these properties in detail and showed, in particular, that our definition naturally leads to the existence of "information paths". We also discussed how information flow can be inferred through conditional independence testing, and provided an algorithm for recovering the information paths in a given system. Finally, we studied some canonical examples of computational systems from different contexts, and showed that our definition of information flow is intuitive in each case.

We proceed to discuss several important assumptions and simplifications in our model. We also discuss existing literature related to estimation of causal influence in neuroscience, and how our computational system model leads us to a significantly different measure of information flow. Similarly, we discuss how our framework is very different from the field of Probabilistic Graphical Models.

A. Neuroscientific Concerns

1) Observing edges vs. nodes: The observation model stated in Section V-A makes a crucial assumption, namely, that transmissions on each *edge* can be observed. In neuroscientific experiments, however, we often record activity from single neurons (as in the case of electrophysiological recordings), or aggregate activity from groups of neurons (as with Local Field Potentials measured in Electrocorticography and Electroencephalography). These neurons, or groups of neurons, are considered to be nodes communicating to one another in a network. It may not be known which nodes are connected to which other nodes, let alone the recipient of each transmission at every time instant. This is a marked departure from our assumption that transmissions on edges can be observed. To some extent, it is possible to incorporate a "node-centric" model within our computational system by assuming that all nodes broadcast their transmissions. However, that still leaves unanswered the question of which nodes actually "hear" another's transmissions. A possible resolution to that question might arise from an understanding of *receiver response*. That is, we consider a revised model in which an edge exists if a receiving neuron *uses* the information transmitted by some neuron at the previous time instant. This issue is beyond the scope of the current work, and will be addressed in subsequent studies.

We also note that, although tools based on Granger Causality implicitly assume that nodes are measured and not edges, they do not resolve the issue of which node is "talking" to which other node. For example, if two different nodes A_1 and B_1 communicate the same information to a third node, C_2, any regression based analysis will assign a weight of one-half to each of A_1 and B_1. However, the true function, f_{C_2}, may be using only the information coming from A_1, or only the information coming from B_1, or using the two in some other unequal proportion. Such cases may only be identifiable through an interventional approach.

Conversely, our work may suggest to neuroscientists that inferences about information flow are more reliably obtained if one can measure transmissions on edges in the graph, rather

than transmissions of nodes. This may call for newer imaging modalities, or new uses of existing modalities, such as treating axons as targets for invasive recordings, perhaps at nodes of Ranvier. Some work along these lines has already appeared: Patolsky et al. [90] develop new techniques to measure signals along the length of an axon. Such ideas may need to be revisited in greater detail, given the importance of measuring edges.

2) Observing memories: Another important assumption in the observation model is that *memories* of nodes are observed as transmissions on self-edges. If these transmissions are implemented in the form of some internal state at each node, then they might be difficult to observe in practice.[27]

It remains to be fully understood whether one can compensate for not observing memories in some manner, e.g., by assuming that the memory of a node is the full history of its transmissions and receptions. While this means that intrinsically generated random variables that are *not* propagated to other nodes will never be observed, it could be argued that such variables could have no impact on the system (save for acting as "computational noise"). So perhaps it suffices to observe only transmissions between *different* nodes (and not self-edges). Further work is required to understand what ramifications such an assumption has on identifying information flows and information paths.

Alternatively, perhaps if one wishes to observe memories, it is important to measure not only spikes, but also membrane voltages (e.g. using voltage-sensitive dyes [91] or, less directly, through measurements of changes in neurotransmitter concentrations outside a cell [92]).

3) Discretization of time: Yet another implicit assumption in our computational system model is that transmissions occur at discrete points in time. This assumption is justified for synchronous digital circuits used commonly today, or if the computational system of interest is a trained artificial neural network, for instance. However, this is not a perfect model of the brain, because neural spiking (among other processes), does not occur only at multiples of some fundamental unit of time. The same also holds true of dendritic and axonal propagation delays, for instance.

This issue might be partially mitigated by assuming that neural computation happens at a certain time scale, and by using a sufficiently high sampling rate so that Nyquist-rate–type arguments apply. However, Nyquist-rate sampling may not be possible in certain modalities that are inherently slow (e.g. Calcium imaging and functional Magnetic Resonance Imaging), so it would be interesting to understand what inferences we are no longer capable of making. Alternatively, if the sampling rate is too high, it may be useful to look for M-information flow within *time windows*, which could help increase the sample size for detection. The exact implications of using such preprocessing methods will also need to be studied in greater detail, and forms another avenue for future work.

[27]If every node represents a *group* of neurons, however it may just be that their internal state is represented in the form of *communication between these neurons*. In that case, perhaps observing their internal state is just a matter of having more spatially refined measurements.

4) Message enters at $t = 0$: Another assumption in our framework is that the message enters the system at, and only at, time $t = 0$. This is essential, given the way we have defined input nodes: nodes at time $t = 0$, whose outputs depend on the message (and which have no other shared source of randomness). However, this assumption does not allow for a dynamically evolving stimulus, which is also common in neuroscientific experiments.

Suppose we allow the message to enter the system at a later time instant, say at some node U_t, for $t > 0$, i.e., U_t may compute a function not just of its inputs, but also of M. Then, if we want the information path theorem to continue to hold, we must also add U_t to the set of input nodes.[28] Thus, if we see dependence at some other node $V_{t'}$, at a later time instant $t' > t$, the information paths leading to $V_{t'}$ may arise from the original input nodes *or* from U_t, or both. As we might intuitively expect, the more time points we allow the message to enter *at*, the more such information paths we will likely see, making the results of our analysis harder to interpret.

On a related note, recall that the assumption that M enters only at $t = 0$ comes from our decision to focus on event related experimental paradigms [28] (refer Section I-B). However, although neural responses to event related stimuli are often time-locked as well, they have considerable variability: i.e., a neuron may respond at random times in each trial. Apart from the inherent stochasticity in neural firing, this could happen for any number of reasons, including the animal's state of arousal, its attention levels in each trial, etc. In our framework, the information flow will be smeared out over the entire time interval of possible response. A modified computational model may be needed to address such an issue.

5) Experimental design and the message: An important aspect of our work is that it explicitly incorporates the message, which in neuroscientific experiments is often some information contained in the stimulus. This aids the neuroscientist in designing experiments, for example, in understanding what stimuli will help them make a certain inference about information flow. In particular, one needs to use at least two different stimuli in order to obtain any determination about information flow. While this is implicitly understood in neuroscience, as evidenced by comparisons with baselines, or by the use of permutation tests to scramble stimulus-trial correlations for a null model, our framework provides a more direct method for identifying and interpreting stimulus-related information flow. This could be particularly useful when tracking the flows of multiple messages individually (see Section V-F).

B. The Difficulty of Estimation

A strategy for detecting edges that have M-information flow was presented in Section V-B. In practice, however, there are several issues associated with employing such a strategy. These are discussed below.

Firstly, we currently assume that observations are noiseless (see Section V-A, Assumption 3). It is unclear, exactly, to what extent noisy observations will impact the inference

[28]We should also expect that any Local Markovity conditions at time t (see Proposition 6) that involve the node U_t will no longer hold.

of information flow. In particular, it is worth understanding whether small amounts of observation noise can be tolerated if all edges with M-information flow have a sufficiently large "volume" of information (i.e., the corresponding mutual or conditional mutual information is sufficiently large). As was described intuitively in Section V-B, if the information volume is large, then even under noisy conditions, we might expect the test statistic to clear the threshold, so the presence of M-information flow can still be detected consistently. But small volumes of information that aggregate over time—e.g. information about M "trickling" over time from one node to another—could still pose issues. Such M-information flow could go undetected, as has been shown to occur in other contexts [55], using a different measure of flow. It is possible that Derived Information, in particular, is hard to infer in the presence of noise. This could make the task of detecting the presence of a hidden node difficult (consider the case of a "trickling" hidden node), as well as that of identifying redundant links.

Secondly, detecting whether each edge at time t has information flow involves checking *all* subsets of $\mathscr{C}_t$. For N nodes and N^2 edges, this implies 2^{N^2} subsets of edges that need to be searched. This could be seen as being prohibitively difficult for $N^2 \geq 30$, or for N greater than about 5 or 6 nodes. However, in reality, graphs in neuroscience are often known to be edge-sparse [93]–[95]. For example, in the brain, a well-established 11-node network is the reward network [95]. Most nodes in this network typically have just one incoming and one outgoing connection. The two most important nodes have five incoming edges each, with two and four outgoing edges respectively. Further, it is known which connections are inhibitory and which are excitatory, which could further help with testing for information flow. A fully connected network would have had 121 edges, but the underlying connectivity of the circuit only allows for a total of 17 edges in this network. So in reality, anatomical priors help reduce the number of edges to well within the range of what is computable. Nevertheless, it remains of interest to find methods by which nodes and/or edges can be excluded from the search, and this could be another topic for further research.

Another statistical issue that crops up when attempting to simultaneously perform several conditional independence tests is the problem of *multiple comparisons* [96]. Simply put, when performing a large number of independent hypotheses tests, say N, at some fixed false alarm rate α, on average, we should expect αN of these tests to erroneously reject the null. In the context of information flow, we might wish to set the null hypothesis to be the absence of M-information flow on a given edge. Then, to test for M-information flow on this edge, we need to perform a large number of conditional independence tests—call this number N—at some false alarm rate α. These tests are, in fact, *not independent* of one another; nevertheless, very loosely put, if we choose a false alarm rate $\alpha \approx 1/N$, we may find that the probability of *at least one* false alarm is too high. This would make us erroneously infer that this particular edge has M-information flow; moreover, since this argument applies to any edge, if α is not chosen conservatively enough, we may erroneously infer that *all* edges have M-information flow.

This multiple hypothesis testing problem is better posed as a "Global Null test" (e.g., see [97]), wherein the global null is the hypothesis that *all* of the conditional independence tests are individually null (i.e., that there is *no* M-information flow on the given edge), and the global alternative is the hypothesis that *at least one* of the conditional independence tests is non-null (i.e., that there *is* M-information flow on the given edge). As mentioned before, however, the conditional independence tests dictated by Definition 4 are, in general, *dependent* on one another. Furthermore, it might not be easy to describe the manner of dependence, so when choosing Global Null tests, it is essential to choose those that work under arbitrary dependence. A simple example of such a test is the well-known Bonferroni correction, which uses a level $\alpha' = \alpha/N$ for each test (where α is the desired false alarm rate for the overall Global Null test); but we may find that such methods have insufficient statistical power. A potential solution to this problem might involve combining multiple Global Null tests in some meaningful way: for example, one could imagine designing a procedure that controls the False Discovery Rate[29] [98] on the *identification of edges* with M-information flow.[30] Another approach might be to find ways of directly testing information *paths*, wherein the hypothesis tested would be that a certain M-information *path* exists in the system, rather than requiring every *edge* with M-information flow be identified first. All of these ideas are potential avenues for future work.

C. The Limitations of Granger Causality and Related Tools

Mapping directed functional connectivity and information flow in the brain has been a hot topic for several years, as evidenced by the large body of work in this direction [37]–[39]. Approaches for statistically mapping functional connectivity often rely on variations of Granger Causality [41] and, more recently, Directed Information [43]–[45], which we here collectively refer to as "Granger Causality-based tools". These approaches lack a systematic framework that ties the statistical analysis to the underlying computation, however, and the interpretations drawn from their use have often been questioned [48], [50], [51], [55]–[58].

In particular, a crucial difference between our approach and that of Granger Causality-based tools is that the latter do not have an explicit description of the message. Instead, they provide mechanisms to condense a pair of time series into a single statistic. There are no concrete models that can be used to interpret what this statistic means for the flow of *information* about the message. Furthermore, if one is interested in the information flow of multiple messages, Granger Causality-based tools do not provide an immediate solution. This is why a tool that ties information flow directly with a message is of great interest to practitioners.

[29]These methods control the expected proportion of false discoveries, i.e., the proportion of null hypotheses that are falsely rejected.

[30]Care is needed when doing this, however, since tests for M-information flow on different edges at the same time instant are *also* dependent on one another.

The absence of an underlying computational framework with well-defined assumptions inherently makes it very hard to draw sound inferences through the application of Granger Causality-based tools. A striking example of this is a recent result of ours [58] that shows, using a feedback communication system, that the direction of greater Granger-causal influence can be opposite to the direction in which the message is communicated, even in the absence of hidden nodes and measurement noise. The time-unrolled graph framework presented here has been specifically designed to address this issue, and present a clear understanding of information flow, even in the presence of feedback. The example given in Section VI-C demonstrates a potential resolution to this issue.

Granger Causality was originally developed for the study of time-series that occur only once, such as in economics [40]. An artifact of this development is that it was not designed to incorporate multiple trials of the same process. Instead, it assumes stationarity to help estimate parameters of the random variables that control the process. In the neuroscientific context, stationarity is often a very poor assumption, since the segment of time-series data corresponding to each trial may be short, and often sees some kind of stimulus presentation. Naturally, presentation of the stimulus changes the underlying parameters of the time-series and destroys stationarity; indeed, this is the quintessential aspect of the experiment. Thus, in order to understand processing in such stimulus-driven tasks, one needs to be able to infer time-dependent information flows from data. While information-theoretic extensions of Granger Causality such as Transfer Entropy and Directed Information do not assume stationarity, they nevertheless fail to provide a dynamically evolving picture of information flow. Recently, some work has started to analyze an "adaptive" form of Granger Causality computed on windowed time intervals [60]; such ideas may be worth pursuing for attaining a dynamically evolving picture of Granger causal influence, though they will still not comprise information flow, due to the absence of a connection to the computational task and the message.

In Section I-B, we discussed two dominant interpretations of "information flow" in neuroscience: the first has to do with information about a specific message and is what we address in this paper. The second, having to do with information in the abstract, is more akin to what is done by Granger Causality-based tools. It may be possible, in some settings and under suitable assumptions, to unite these two interpretations. For instance, it would be useful to know under what conditions (e.g., Gaussianity, linear functions, etc.), Granger causal influence provides the same inferences as our rigorous notion of M-information flow. This is a promising future direction, since it is important to understand in which situations Granger Causality-based methods recover meaningful flows of information, and in which cases we must be careful with interpretation.

D. Probabilistic Graphical Models and Pearl's Causality

There is one important difference that distinguishes our work from the perspective adopted in the field of probabilistic graphical models (PGMs) [99], and the representations therein. In our framework, nodes represent computational units, whereas in PGMs, nodes represent the random variables themselves, and edges capture the conditional independence relationships between these variables. While it might be possible to construct a PGM that is equivalent to our computational model, this would likely eliminate any intuitive structure captured by the computational graph.

It remains to be understood whether and how Pearl's notions of causality [30] can be seamlessly merged with the understanding of information flow developed here. We expect that some formal application of causality will be needed in going from an edge-centric model (as presented here) to a more node-centric one (discussed in Section VII-A), in order to identify which transmissions influenced a given node's output.

There are several works in the literature that discuss measures of information flow in probabilistic graphical models [100], [101], but they are heavily inspired by causality and largely center around an interventionist approach. In contrast, our definition of information flow is based on a computational system model that translates more readily to neuroscience, and we assume that the experimentalist is restricted to making observations. Nevertheless, it may still be interesting to explore alternative definitions of information flow, which incorporate interventional or counterfactual reasoning. Understanding the connection between information flow and interventional approaches could be essential for clinical translation, and constitutes another important direction for future work.

E. Future Directions for Theoretical Development

A natural question that arises from this paper is: how can our definition of information flow on an edge be extended to a more generic information *measure*, which also quantifies the volume of flow? Finding such a measure will involve aggregating the conditional mutual information for each subset of edges into a single value (one example of such a measure was provided in Section VI-C, though it was not developed from first-principles). It is as yet unclear how this might be achieved, while still gelling well with our intuition of what this information flow volume ought to be. We believe that the right approach is to start by designating a set of properties that we would like information flow volumes to satisfy, and then to propose a measure through the use of representative examples and counterexamples.

A second direction that emerges is related to Partial Information Decomposition (PID) [65]–[67], which was discussed earlier in Section III-E. M-information flow is very closely related to the PID: while Candidate Definition 1 checks for positivity of mutual information between M and $X(E_t)$, and hence implies the presence of unique and/or redundant information, our definition also detects the presence of purely synergistic information. Since our definition is closely tied to computation and is strongly motivated through the goal of finding unbroken information paths, the close relationship between PID and our definition suggests that PID might be the right toolset for obtaining a more fine-grained understanding of information flow. In particular, it would be useful to know how our understanding of information representation and computation is enhanced through a PID analysis (we try to take this

approach in some very preliminary work on error correction in grid cells [102]). Finally, we note that the PID could also help inform the discussion on a definition for information volume. Providing a useful definition of information volume based on current definitions of unique, redundant and synergistic information, and asking whether the problem of information flow can inform the PID literature, will also be the subject of future research.

A third direction has to do with alternate definitions of information flow: there might be other definitions of information about a message, which satisfy the information path property. These could be arrived at through modifications to our current definition, or by looking at directions we did not pursue here, e.g., counterfactual measures. It is worth understanding whether such definitions can avoid M-information orphans, or whether there will be more counterexamples to the use of such measures (we recently made some forays along these lines [103]). Furthermore, the properties we stated in this paper are not sufficient to uniquely specify our definition of information flow. For example, the all-zero function as well as the all-ones function satisfy the Broken Telephone property, although they are not particularly useful definitions of information flow. Thus, it would be useful to understand what other properties we should impose so as to arrive at a unique definition of information flow. As a crude and preliminary example, we demonstrate how this might be done in Appendix E.

F. Concluding Remarks

We conclude by describing some of our general impressions in working on the theoretical development presented in this paper. As such, these points merely highlight some of our opinions on how theory—and more specifically, information theory—may be applied in neuroscience.

As mentioned in the introduction, we drew inspiration from two papers that discuss how experimentalists understand systems in biology and neuroscience [32], [33]. Both these works advocate for theory by arguing that we need new analytical tools, and that the accumulation of empirical knowledge alone does not constitute *understanding*. Lazebnik [32], in particular, mentions how terminology in biology tends to be vague and non-committal. We feel that an important reason for this is the absence of concrete underlying models, with clearly-stated assumptions. In other words, we think that theory and modeling can go a long way in providing a *language* that will enable well-grounded discussions. This language, in turn, arises through the development of theoretical models and formal definitions.

Another point made by both the aforementioned papers is that we should attempt to understand large computational systems by first examining smaller models, and models in which the ground truth is already known. This approach allows us to create new analytical tools that can be thoroughly vetted, so that the interpretations drawn from their use in experimental practice is unambiguous and undebated. We also believe that when trying to understand large computational systems, it is essential to start with toy models such as Counterexample 1. This philosophy of starting with toy models, and abstracting out meaningful ideas that hold more generally in large systems, is well-entrenched in the field of information theory, and can become a useful export in fields such as neuroscience.

Acknowledgments

We have many people to thank for extremely useful discussions. A non-exhaustive list follows: Mayank Bakshi, Marlene Behrmann, Todd Coleman, Elliot Collins, Uday Jagadisan, Haewon Jeong, Rob Kass, José Moura, Bobak Nazer, Venkatesh Saligrama, Gabe Schamberg, Tsachy Weissman. We also thank the anonymous reviewers whose comments improved our exposition substantially.

Praveen Venkatesh was supported, in part, by a Fellowship in Digital Health from the Center for Machine Learning and Health at Carnegie Mellon University and by a Dowd Fellowship from the College of Engineering at Carnegie Mellon University. The authors would like to thank Philip and Marsha Dowd for their financial support and encouragement. Sanghamitra Dutta was supported, in part, by a K&L Gates Presidential Fellowship in Ethics and Computational Technologies. Pulkit Grover was supported, in part, by an NSF CAREER Award.

References

[1] P. Venkatesh, S. Dutta, and P. Grover, "How should we define information flow in neural circuits?" in *2019 IEEE International Symposium on Information Theory (ISIT)*, July 2019, pp. 176–180.

[2] J. Almeida, A. R. Fintzi, and B. Z. Mahon, "Tool manipulation knowledge is retrieved by way of the ventral visual object processing pathway," *Cortex*, vol. 49, no. 9, pp. 2334–2344, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0010945213001329

[3] Y. K. Hong, C. O. Lacefield, C. C. Rodgers, and R. M. Bruno, "Sensation, movement and learning in the absence of barrel cortex," *Nature*, vol. 561, no. 7724, pp. 542–546, 2018.

[4] A. Brovelli, M. Ding, A. Ledberg, Y. Chen, R. Nakamura, and S. L. Bressler, "Beta oscillations in a large-scale sensorimotor cortical network: Directional influences revealed by Granger causality," *Proceedings of the National Academy of Sciences*, vol. 101, no. 26, pp. 9849–9854, 2004. [Online]. Available: http://www.pnas.org/content/101/26/9849

[5] M. Bar, K. S. Kassam, A. S. Ghuman, J. Boshyan, A. M. Schmid, A. M. Dale, M. S. Hämäläinen, K. Marinkovic, D. L. Schacter, B. R. Rosen, and E. Halgren, "Top-down facilitation of visual recognition," *Proceedings of the National Academy of Sciences*, vol. 103, no. 2, pp. 449–454, 2006. [Online]. Available: https://www.pnas.org/content/103/2/449

[6] A. S. Greenberg, T. Verstynen, Y.-C. Chiu, S. Yantis, W. Schneider, and M. Behrmann, "Visuotopic cortical connectivity underlying attention revealed with white-matter tractography," *Journal of Neuroscience*, vol. 32, no. 8, pp. 2773–2782, 2012. [Online]. Available: http://www.jneurosci.org/content/32/8/2773

[7] R. Ahlswede, N. Cai, S. Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Transactions on Information Theory*, vol. 46, no. 4, pp. 1204–1216, July 2000.

[8] M. Franceschetti and R. Meester, *Random networks for communication: from statistical physics to information systems*. Cambridge University Press, 2008, vol. 24.

[9] P. Grover, "Actions can speak more clearly than words," Ph.D. dissertation, University of California, Berkeley, 2010.

[10] G. Ranade, "Active systems with uncertain parameters: an information-theoretic perspective," Ph.D. dissertation, University of California, Berkeley, 2014.

[11] G. Smith, "On the foundations of quantitative information flow," in *International Conference on Foundations of Software Science and Computational Structures*. Springer, 2009, pp. 288–302.

[12] A. Datta, S. Sen, and Y. Zick, "Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems," in *2016 IEEE symposium on security and privacy (SP)*. IEEE, 2016, pp. 598–617.

[13] S. Dutta, P. Venkatesh, P. Mardziel, A. Datta, and P. Grover, "An information-theoretic quantification of discrimination with exempt features," in *Proceedings of the Thirty-fourth AAAI Conference on Artificial Intelligence*, 2020, (to appear).

[14] Z. Goldfeld, E. v. d. Berg, K. Greenewald, I. Melnyk, N. Nguyen, B. Kingsbury, and Y. Polyanskiy, "Estimating information flow in deep neural networks," in *Proceedings of the 36th International Conference on Machine Learning*, 2019, pp. 2299–2308.

[15] N. Tishby, F. C. Pereira, and W. Bialek, "The information bottleneck method," *arXiv:physics/0004057 [physics.data-an]*, 2000.

[16] N. Tishby and N. Zaslavsky, "Deep learning and the information bottleneck principle," in *2015 IEEE Information Theory Workshop (ITW)*. IEEE, 2015, pp. 1–5.

[17] M. Newman, *Networks*. Oxford University Press, 2018.

[18] Y. LeCun, J. S. Denker, and S. A. Solla, "Optimal brain damage," in *Advances in Neural Information Processing Systems*, 1990, pp. 598–605.

[19] J. Frankle and M. Carbin, "The lottery ticket hypothesis: Finding sparse, trainable neural networks," in *7th International Conference on Learning Representations, ICLR 2019*, 2019. [Online]. Available: https://openreview.net/forum?id=rJl-b3RcF7

[20] C. Hammond, H. Bergman, and P. Brown, "Pathological synchronization in Parkinson's disease: networks, models and treatments," *Trends in neurosciences*, vol. 30, no. 7, pp. 357–364, 2007.

[21] Y. Smith, M. Bevan, E. Shink, and J. Bolam, "Microcircuitry of the direct and indirect pathways of the basal ganglia," *Neuroscience*, vol. 86, no. 2, pp. 353–387, 1998.

[22] A. A. Grace, "Gating of information flow within the limbic system and the pathophysiology of schizophrenia," *Brain Research Reviews*, vol. 31, no. 2-3, pp. 330–341, 2000.

[23] E. Lalo, S. Thobois, A. Sharott, G. Polo, P. Mertens, A. Pogosyan, and P. Brown, "Patterns of bidirectional communication between cortex and basal ganglia during movement in patients with Parkinson disease," *Journal of Neuroscience*, vol. 28, no. 12, pp. 3008–3016, 2008.

[24] E. H. Kossoff, E. K. Ritzl, J. M. Politsky, A. M. Murro, J. R. Smith, R. B. Duckrow, D. D. Spencer, and G. K. Bergey, "Effect of an external responsive neurostimulator on seizures and electrographic discharges during subdural electrode monitoring," *Epilepsia*, vol. 45, no. 12, pp. 1560–1567, 2004.

[25] A. L. Benabid, "Deep brain stimulation for Parkinson's disease," *Current opinion in neurobiology*, vol. 13, no. 6, pp. 696–706, 2003.

[26] G. Clark, "Cochlear implants," in *Speech processing in the auditory system*. Springer, 2004, pp. 422–462.

[27] J. D. Weiland, W. Liu, and M. S. Humayun, "Retinal prosthesis," *Annu. Rev. Biomed. Eng.*, vol. 7, pp. 361–401, 2005.

[28] J. Samuels and N. D. Zasler, *Event-Related Paradigms*. Springer, 2018, pp. 1346–1347.

[29] C. D. Thompson, "A complexity theory for VLSI," Ph.D. dissertation, Carnegie Mellon University, Pittsburgh, PA, USA, 1980, aAI8100621.

[30] J. Pearl, *Causality: Models, Reasoning and Inference*. Cambridge University Press, 2009.

[31] J. Peters, D. Janzing, and B. Schölkopf, *Elements of causal inference: Foundations and learning algorithms*. MIT press, 2017.

[32] Y. Lazebnik, "Can a biologist fix a radio?—Or, what I learned while studying apoptosis," *Cancer cell*, vol. 2, no. 3, pp. 179–182, 2002.

[33] E. Jonas and K. P. Kording, "Could a neuroscientist understand a microprocessor?" *PLoS computational biology*, vol. 13, no. 1, p. e1005268, 2017.

[34] P. Grover and P. Venkatesh, "An information-theoretic view of EEG sensing," *Proceedings of the IEEE*, vol. 105, no. 2, pp. 367–384, Feb 2017.

[35] P. Grover, J. A. Weldon, S. K. Kelly, P. Venkatesh, and H. Jeong, "An information theoretic technique for harnessing attenuation of high spatial frequencies to design ultra-high-density EEG," in *2015 53rd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, Sept 2015, pp. 901–908.

[36] P. Venkatesh and P. Grover, "Lower bounds on the minimax risk for the source localization problem," in *2017 IEEE International Symposium on Information Theory (ISIT)*, June 2017, pp. 3080–3084.

[37] K. J. Friston, "Functional and effective connectivity: a review," *Brain connectivity*, vol. 1, no. 1, pp. 13–36, 2011.

[38] K. Friston, R. Moran, and A. K. Seth, "Analysing connectivity with Granger causality and dynamic causal modelling," *Current opinion in neurobiology*, vol. 23, no. 2, pp. 172–178, 2013.

[39] A. M. Bastos and J.-M. Schoffelen, "A tutorial review of functional connectivity analysis methods and their interpretational pitfalls," *Frontiers in systems neuroscience*, vol. 9, p. 175, 2016.

[40] C. W. J. Granger, "Investigating causal relations by econometric models and cross-spectral methods," *Econometrica: Journal of the Econometric Society*, pp. 424–438, 1969.

[41] S. L. Bressler and A. K. Seth, "Wiener–Granger causality: A well established methodology," *NeuroImage*, vol. 58, no. 2, pp. 323–329, 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1053811910002272

[42] J. Massey, "Causality, feedback and directed information," in *Proceedings of the International Symposium on Information Theory and its Applications (ISITA)*, 1990, pp. 303–305.

[43] C. J. Quinn, T. P. Coleman, N. Kiyavash, and N. G. Hatsopoulos, "Estimating the directed information to infer causal relationships in ensemble neural spike train recordings," *Journal of Computational Neuroscience*, vol. 30, no. 1, pp. 17–44, Feb 2011. [Online]. Available: https://doi.org/10.1007/s10827-010-0247-2

[44] C. J. Quinn, N. Kiyavash, and T. P. Coleman, "Directed information graphs," *IEEE Transactions on Information Theory*, vol. 61, no. 12, pp. 6887–6909, Dec 2015.

[45] J. Jiao, H. H. Permuter, L. Zhao, Y. Kim, and T. Weissman, "Universal estimation of directed information," *IEEE Transactions on Information Theory*, vol. 59, no. 10, pp. 6220–6242, Oct 2013.

[46] T. Schreiber, "Measuring information transfer," *Physical Review Letters*, vol. 85, pp. 461–464, Jul 2000. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevLett.85.461

[47] L. A. Baccalá and K. Sameshima, "Partial directed coherence: a new concept in neural structure determination," *Biological Cybernetics*, vol. 84, no. 6, pp. 463–474, May 2001. [Online]. Available: https://doi.org/10.1007/PL00007990

[48] O. David, I. Guillemain, S. Saillet, S. Reyt, C. Deransart, C. Segebarth, and A. Depaulis, "Identifying neural drivers with functional MRI: an electrophysiological validation," *PLoS biology*, vol. 6, no. 12, p. e315, 2008.

[49] A. Roebroeck, E. Formisano, and R. Goebel, "The identification of interacting networks in the brain using fMRI: model selection, causality and deconvolution," *Neuroimage*, vol. 58, no. 2, pp. 296–302, 2011.

[50] O. David, "fMRI connectivity, meaning and empiricism. comments on: Roebroeck et al. The identification of interacting networks in the brain using fMRI: model selection, causality and deconvolution." *Neuroimage*, vol. 58, no. 2, pp. 306–309, 2011.

[51] P. A. Stokes and P. L. Purdon, "A study of problems encountered in Granger causality analysis from a neuroscience perspective," *Proceedings of the National Academy of Sciences*, vol. 114, no. 34, pp. E7063–E7072, 2017. [Online]. Available: http://www.pnas.org/content/114/34/E7063

[52] L. Barnett, A. B. Barrett, and A. K. Seth, "Solved problems for Granger causality in neuroscience: A response to Stokes and Purdon," *NeuroImage*, vol. 178, pp. 744–748, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1053811918304932

[53] L. Faes, S. Stramaglia, and D. Marinazzo, "On the interpretability and computational reliability of frequency-domain Granger causality," *F1000Research*, Sep 2017.

[54] P. A. Stokes and P. L. Purdon, "In reply to Faes et al. and Barnett et al. regarding "a study of problems encountered in Granger causality analysis from a neuroscience perspective"," *arXiv:1709.10248 [stat.ME]*, 2017.

[55] J. Andersson, "Testing for Granger causality in the presence of measurement errors," *Economics Bulletin*, 2005.

[56] H. Nalatore, M. Ding, and G. Rangarajan, "Mitigating the effects of measurement noise on Granger causality," *Physical Review E*, vol. 75, no. 3, p. 031123, Mar 2007. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevE.75.031123

[57] M. Gong, K. Zhang, B. Schölkopf, D. Tao, and P. Geiger, "Discovering temporal causal relations from subsampled data," in *Proceedings of The 32nd International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 37. PMLR, Jul 2015, pp. 1898–1906. [Online]. Available: http://proceedings.mlr.press/v37/gongb15.html

[58] P. Venkatesh and P. Grover, "Is the direction of greater Granger causal influence the same as the direction of information flow?" in *53rd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, Sept 2015, pp. 672–679.

[59] C. E. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, July 1948.

[60] A. Sheikhattar, S. Miran, J. Liu, J. B. Fritz, S. A. Shamma, P. O. Kanold, and B. Babadi, "Extracting neuronal functional network dynamics via adaptive Granger causality analysis," *Proceedings of the National Academy of Sciences*, vol. 115, no. 17, pp. E3869–E3878, 2018.

[61] K. J. Friston, L. Harrison, and W. Penny, "Dynamic causal modelling," *Neuroimage*, vol. 19, no. 4, pp. 1273–1302, 2003.

[62] A. V. Oppenheim, J. R. Buck, and R. W. Schafer, *Discrete-time signal processing*, 2nd ed. Upper Saddle River, N.J.: Prentice Hall, 1999.

[63] Y. Polyanskiy and Y. Wu, "Lecture notes on information theory," August 2017. [Online]. Available: http://www.stat.yale.edu/~yw562/teaching/itlectures.pdf

[64] C. E. Shannon, "Communication theory of secrecy systems," *Bell system technical journal*, vol. 28, no. 4, pp. 656–715, 1949.

[65] P. L. Williams and R. D. Beer, "Nonnegative decomposition of multivariate information," *arXiv:1004.2515 [cs.IT]*, 2010.

[66] M. Harder, C. Salge, and D. Polani, "Bivariate measure of redundant information," *Phys. Rev. E*, vol. 87, p. 012130, Jan 2013. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevE.87.012130

[67] N. Bertschinger, J. Rauh, E. Olbrich, J. Jost, and N. Ay, "Quantifying unique information," *Entropy*, vol. 16, no. 4, pp. 2161–2183, 2014. [Online]. Available: http://www.mdpi.com/1099-4300/16/4/2161

[68] J. T. Lizier, N. Bertschinger, J. Jost, and M. Wibral, "Information decomposition of target effects from multi-source interactions: Perspectives on previous, current and future work," *Entropy*, vol. 20, no. 4, p. 307, 2018.

[69] E. Schneidman, W. Bialek, and M. J. Berry, "Synergy, redundancy, and independence in population codes," *Journal of Neuroscience*, vol. 23, no. 37, pp. 11 539–11 553, 2003.

[70] P. E. Latham and S. Nirenberg, "Synergy, redundancy, and independence in population codes, revisited," *Journal of Neuroscience*, vol. 25, no. 21, pp. 5195–5206, 2005.

[71] N. M. Timme and C. Lapish, "A tutorial for information theory in neuroscience," *eNeuro*, vol. 5, no. 3, 2018.

[72] I. Gat and N. Tishby, "Synergy and redundancy among brain cells of behaving monkeys," in *Advances in neural information processing systems*, 1999, pp. 111–117.

[73] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. John Wiley & Sons, 2012.

[74] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 3rd ed. The MIT Press, 2009.

[75] W. P. Bergsma, "Testing conditional independence for continuous random variables," *EURANDOM report*, vol. 2004, no. 049, 2004.

[76] K. Zhang, J. Peters, D. Janzing, and B. Schölkopf, "Kernel-based conditional independence test and application in causal discovery," in *Proceedings of the Twenty-Seventh Conference on Uncertainty in Artificial Intelligence*, ser. UAI'11. Arlington, Virginia, United States: AUAI Press, 2011, pp. 804–813. [Online]. Available: http://dl.acm.org/citation.cfm?id=3020548.3020641

[77] T.-M. Huang *et al.*, "Testing conditional independence using maximal nonlinear conditional correlation," *The Annals of Statistics*, vol. 38, no. 4, pp. 2047–2091, 2010.

[78] L. Su and H. White, "A consistent characteristic function-based test for conditional independence," *Journal of Econometrics*, vol. 141, no. 2, pp. 807–834, 2007. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0304407606002375

[79] M. Huang, Y. Sun, and H. White, "A flexible nonparametric test for conditional independence," *Econometric Theory*, vol. 32, no. 6, pp. 1434–1482, 2016.

[80] R. Sen, K. Shanmugam, H. Asnani, A. Rahimzamani, and S. Kannan, "Mimic and classify: A meta-algorithm for conditional independence testing," 2018.

[81] R. D. Shah and J. Peters, "The hardness of conditional independence testing and the generalised covariance measure," *arXiv:1804.07203 [math.ST]*, Apr 2018. [Online]. Available: https://arxiv.org/abs/1804.07203

[82] L. Paninski, "Estimation of entropy and mutual information," *Neural computation*, vol. 15, no. 6, pp. 1191–1253, 2003.

[83] W. Gao, S. Kannan, S. Oh, and P. Viswanath, "Estimating mutual information for discrete-continuous mixtures," in *Advances in Neural Information Processing Systems*, 2017, pp. 5986–5997.

[84] A. Kraskov, H. Stögbauer, and P. Grassberger, "Estimating mutual information," *Physical review E*, vol. 69, no. 6, p. 066138, 2004.

[85] H. Liu, L. Wasserman, and J. D. Lafferty, "Exponential concentration for mutual information estimation with application to forests," in *Advances in Neural Information Processing Systems*, 2012, pp. 2537–2545.

[86] J.-P. Lachaux, E. Rodriguez, J. Martinerie, and F. J. Varela, "Measuring phase synchrony in brain signals," *Human brain mapping*, vol. 8, no. 4, pp. 194–208, 1999.

[87] F. Varela, J.-P. Lachaux, E. Rodriguez, and J. Martinerie, "The brainweb: phase synchronization and large-scale integration," *Nature reviews neuroscience*, vol. 2, no. 4, p. 229, 2001.

[88] M. I. Posner, "Orienting of attention," *Quarterly journal of experimental psychology*, vol. 32, no. 1, pp. 3–25, 1980.

[89] J. Schalkwijk and T. Kailath, "A coding scheme for additive noise channels with feedback–I: No bandwidth constraint," *Information Theory, IEEE Transactions on*, vol. 12, no. 2, pp. 172–182, 1966.

[90] F. Patolsky, B. P. Timko, G. Yu, Y. Fang, A. B. Greytak, G. Zheng, and C. M. Lieber, "Detection, stimulation, and inhibition of neuronal signals with high-density nanowire transistor arrays," *Science*, vol. 313, no. 5790, pp. 1100–1104, 2006.

[91] A. Grinvald and R. Hildesheim, "VSDI: a new era in functional imaging of cortical dynamics," *Nature Reviews Neuroscience*, vol. 5, no. 11, p. 874, 2004.

[92] C. J. Watson, B. J. Venton, and R. T. Kennedy, "In vivo measurements of neurotransmitters by microdialysis sampling," *Analytical Chemistry*, vol. 78, no. 5, pp. 1391–1399, Mar 2006. [Online]. Available: https://doi.org/10.1021/ac0693722

[93] D. S. Bassett and E. Bullmore, "Small-world brain networks," *The Neuroscientist*, vol. 12, no. 6, pp. 512–523, 2006.

[94] S. Achard, R. Salvador, B. Whitcher, J. Suckling, and E. Bullmore, "A resilient, low-frequency, small-world human brain functional network with highly connected association cortical hubs," *Journal of Neuroscience*, vol. 26, no. 1, pp. 63–72, 2006.

[95] S. J. Russo and E. J. Nestler, "The brain reward circuitry in mood disorders," *Nature Reviews Neuroscience*, vol. 14, no. 9, p. 609, 2013.

[96] J. P. Shaffer, "Multiple hypothesis testing," *Annual review of psychology*, vol. 46, no. 1, pp. 561–584, 1995.

[97] B. Duan, A. Ramdas, S. Balakrishnan, and L. Wasserman, "Interactive martingale tests for the global null," *arXiv:1909.07339 [stat.ME]*, 2019.

[98] Y. Benjamini and Y. Hochberg, "Controlling the false discovery rate: A practical and powerful approach to multiple testing," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 57, no. 1, pp. 289–300, 1995. [Online]. Available: http://www.jstor.org/stable/2346101

[99] D. Koller, N. Friedman, and F. Bach, *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[100] N. Ay and D. Polani, "Information flows in causal networks," *Advances in Complex Systems*, vol. 11, no. 01, pp. 17–41, 2008. [Online]. Available: https://doi.org/10.1142/S0219525908001465

[101] D. Janzing, D. Balduzzi, M. Grosse-Wentrup, and B. Schölkopf, "Quantifying causal influences," *Ann. Statist.*, vol. 41, no. 5, pp. 2324–2358, 10 2013. [Online]. Available: https://doi.org/10.1214/13-AOS1145

[102] P. Venkatesh and P. Grover, "Understanding encoding and redundancy in grid cells using partial information decomposition," in *Computational and Systems Neuroscience (Cosyne)*, 2020. [Online]. Available: https://praveenv253.github.io/publications#Venkatesh2020Understanding

[103] P. Venkatesh, S. Dutta, and P. Grover, "How else can we define information flow in neural circuits?" in *2020 IEEE International Symposium on Information Theory (ISIT)*, June 2020, (to appear).

APPENDIX A
PROOF OF PROPOSITION 1

Proof of Proposition 1: ($\Rightarrow$) Suppose there exists some edge $E'_t \in \mathscr{E}'_t$ that has M-information flow as per Definition 4. That is,

$$\exists\, \mathscr{E}''_t \subseteq \mathscr{E}_t \setminus \{E'_t\} \quad \text{s.t.} \quad I\big(M; X(E'_t) \,\big|\, X(\mathscr{E}''_t)\big) > 0. \quad (77)$$

Then,

$$I\big(M; X(\mathscr{E}'_t) \,\big|\, X(\mathscr{E}''_t)\big) = I\big(M; X(E'_t) \,\big|\, X(\mathscr{E}''_t)\big)$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIT.2020.2987806, IEEE Transactions on Information Theory

$$+ I\big(M; X(\mathscr{C}_t' \backslash \{E_t'\}) \,\big|\, X(\mathscr{C}_t''), X(E_t')\big) \tag{78}$$

$$\overset{(a)}{\geq} I\big(M; X(E_t') \,\big|\, X(\mathscr{C}_t'')\big) \overset{(b)}{>} 0 \tag{79}$$

where (a) follows from the non-negativity of conditional mutual information and (b) from (77). Taking $\mathscr{R}_t' \coloneqq \mathscr{C}_t''$ in Definition 5, we see that the set $\mathscr{C}_t'$ has M-information flow.

($\Leftarrow$) Next, suppose that the set $\mathscr{C}_t'$ has M-information flow, as per Definition 5. That is, there exists a set $\mathscr{R}_t' \subseteq \mathscr{C}_t$ such that

$$I\big(M; X(\mathscr{C}_t') \,\big|\, X(\mathscr{R}_t')\big) > 0. \tag{80}$$

Also, let $\{E_t^{(1)}, E_t^{(2)}, \dots E_t^{(K)}\}$ be any ordering of the nodes in $\mathscr{C}_t'$ (where $K = |\mathscr{C}_t'|$). Then by the chain rule of mutual information,

$$0 < I\big(M; X(\mathscr{C}_t') \,\big|\, X(\mathscr{R}_t')\big) \tag{81}$$

$$= \sum_{k=1}^{K} I\Big(M; X(E_t^{(k)}) \,\Big|\, X(\mathscr{R}_t'), X\Big(\bigcup_{j=1}^{k-1} \{E_t^{(j)}\}\Big)\Big). \tag{82}$$

By the non-negativity of conditional mutual information, at least one of the terms in the summation must be strictly positive. Let the index of this term be k^*. Hence, there exists $E_t' \coloneqq E_t^{(k^*)}$ and $\mathscr{C}_t'' \coloneqq \mathscr{R}_t' \cup \{E_t^{(1)}, \dots E_t^{(k^*-1)}\}$, such that

$$I\big(M; X(E_t') \,\big|\, X(\mathscr{C}_t'')\big) > 0. \tag{83}$$

In other words, there exists an edge $E_t' \in \mathscr{C}_t'$ that has M-information flow as per Definition 4. ∎

Appendix B
Proof of Proposition 9

Proof of Proposition 9: Consider the set of all $E_t \in \mathscr{C}_t$ that have M-information flow. That is, E_t must satisfy

$$\exists\, \mathscr{C}_t' \subseteq \mathscr{C}_t \quad \text{s.t.} \quad I\big(M; X(E_t) \,\big|\, X(\mathscr{C}_t')\big) > 0. \tag{84}$$

Define

$$\begin{aligned} \mathscr{R}_t &\coloneqq \{E_t \in \mathscr{C}_t : (84) \text{ holds}\}, \\ \mathscr{S}_t &\coloneqq \mathscr{C}_t \setminus \mathscr{R}_t. \end{aligned} \tag{85}$$

Then, we claim that $\mathscr{R}_t$ and $\mathscr{S}_t$ satisfy equations (63) and (64).

First, note that if $\mathscr{S}_t \neq \emptyset$, then for every $S_t \in \mathscr{S}_t$, we must have that

$$\forall\, \mathscr{C}_t' \subseteq \mathscr{C}_t, \quad I\big(M; X(S_t) \,\big|\, X(\mathscr{C}_t')\big) = 0. \tag{86}$$

If not, then $S_t \in \mathscr{R}_t$ by (85), which implies that $S_t \notin \mathscr{S}_t$, which is a contradiction. Hence, we see that no edge in $\mathscr{S}_t$ has M-information flow. Therefore, by Proposition 1, the set $\mathscr{S}_t$ has no M-information flow. This directly implies the condition in (64).

Next, we claim that if $\mathscr{R}_t \neq \emptyset$, then for every $R_t \in \mathscr{R}_t$, if $\mathscr{C}_t' \subseteq \mathscr{C}_t$ is a set that satisfies

$$I\big(M; X(R_t) \,\big|\, X(\mathscr{C}_t')\big) > 0, \tag{87}$$

then $\mathscr{R}_t' \coloneqq \mathscr{C}_t' \cap \mathscr{R}_t$ satisfies

$$I\big(M; X(R_t) \,\big|\, X(\mathscr{R}_t')\big) > 0. \tag{88}$$

Let $\mathscr{S}_t' \coloneqq \mathscr{C}_t' \setminus \mathscr{R}_t'$, so that $\mathscr{S}_t' \subseteq \mathscr{S}_t$. Then,

$$I\big(M; X(R_t) \,\big|\, X(\mathscr{R}_t'), X(\mathscr{S}_t')\big) > 0 \tag{89}$$

by (87). So,

$$I\big(M; X(R_t) \,\big|\, X(\mathscr{R}_t')\big)$$

$$\overset{(a)}{=} I\big(M; X(R_t), X(\mathscr{S}_t') \,\big|\, X(\mathscr{R}_t')\big) - I\big(M; X(\mathscr{S}_t') \,\big|\, X(\mathscr{R}_t'), X(R_t)\big) \tag{90}$$

$$\overset{(b)}{=} I\big(M; X(R_t), X(\mathscr{S}_t') \,\big|\, X(\mathscr{R}_t')\big) \tag{91}$$

$$\overset{(c)}{=} I\big(M; X(R_t) \,\big|\, X(\mathscr{R}_t'), X(\mathscr{S}_t')\big) + I\big(M; X(\mathscr{S}_t') \,\big|\, X(\mathscr{R}_t')\big) \tag{92}$$

$$\overset{(d)}{=} I\big(M; X(R_t) \,\big|\, X(\mathscr{R}_t'), X(\mathscr{S}_t')\big) \tag{93}$$

$$\overset{(e)}{>} 0,$$

where (a) and (c) follow from the chain rule, (b) and (d) follow from (64), and (e) follows from (89). Thus, condition (63) also holds. ∎

Appendix C
Synergistic Information Flow

A. Partial Information Decomposition preliminaries

The literature on Partial Information Decomposition seeks to find a decomposition for the mutual information between a message, M, and a set of random variables, $\{X_1, X_2, \dots\}$ into several individually meaningful, non-negative terms [68]. For our purposes, it suffices to consider the *bivariate* case, i.e., the decomposition of $I(M; X, Y)$ into non-negative components. In the bivariate case, it is well-understood *how many* components there ought to be, and what these quantities *intuitively represent*, but as yet, there is no consensus on a single set of definitions [68].

There is, however, consensus on a basic set of properties that we expect these components to satisfy. For our purposes, we will only make use of the basic properties stated here, so that *any definition* of the aforementioned components which satisfies these properties suffices for our theory.

In the bivariate case, the mutual information between M and (X, Y) is decomposed into four components: information about M which is (i) unique to X and not present in Y, (ii) unique to Y and not present in X, (iii) redundantly present in both X and Y, and (iv) synergistically present in X and Y. In the notation of [67], the decomposition is written as:

$$\begin{aligned} I\big(M; (X, Y)\big) = {} & UI(M : X \setminus Y) + UI(M : Y \setminus X) \\ & + RI(M : X; Y) + SI(M : X; Y), \end{aligned} \tag{94}$$

where the components are ordered exactly as stated above. Note the distinction between the colon and the semicolon in RI and SI. Also, "$\setminus$" uses the symbol for set-negation to mean "not present in", while also explicitly capturing the asymmetry between X and Y in UI. However, in what follows, we shall assume that RI and SI *are* symmetric in X and Y. This is usually an additional condition that is imposed when defining these quantities, but here, we take it as given.

Given what we want the four components to represent, we would also expect the following to hold:

$$\begin{aligned} I(M; X) &= UI(M : X \setminus Y) + RI(M : X; Y), \\ I(M; Y) &= UI(M : Y \setminus X) + RI(M : X; Y). \end{aligned} \tag{95}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIT.2020.2987806, IEEE Transactions on Information Theory

As a natural consequence, this means that the conditional mutual information will satisfy:

$$\begin{aligned} I(M;X\,|\,Y) &= I\big(M;(X,Y)\big) - I(M;Y) \\ &= UI(M:X\setminus Y) + SI(M:X;Y), \\ I(M;Y\,|\,X) &= I\big(M;(Y,X)\big) - I(M;X) \\ &= UI(M:Y\setminus X) + SI(M:X;Y). \end{aligned} \tag{96}$$

Finally, we want each of these components to always be non-negative:

$$\begin{aligned} UI(M:X\setminus Y)\geq 0 \qquad & RI(M:X;Y)\geq 0 \\ UI(M:Y\setminus X)\geq 0 \qquad & SI(M:X;Y)\geq 0. \end{aligned} \tag{97}$$

It is not obvious that a consistent definition of these four quantities which also satisfies the equations stated above even *exists*, but in fact, additional properties are required to obtain a unique definition. For instance, see [67] for one such development.

As stated before, our theory only relies on the properties stated in this section. As a result, our theorem on the equivalence of information flow definitions holds irrespective of what definition is used, exactly, for synergistic information. It only matters that the definition used satisfies the basic properties presented here.

B. Equivalence of information flow definitions

Proof of Proposition 2: ($\Rightarrow$) Suppose the edge E_t has M-information flow. Then,

$$\exists\ \mathscr{E}_t'\subseteq\mathscr{E}_t \quad \text{s.t.} \quad I\big(M;X(E_t)\,\big|\,X(\mathscr{E}_t')\big)>0. \tag{98}$$

If $I\big(M;X(E_t)\big)>0$ with $\mathscr{E}_t'=\emptyset$ in (98), then condition 1 in Definition 6 holds, so nothing remains to be shown. If not, then $I\big(M;X(E_t)\big)=0$, so (98) implies that there must exist some $\mathscr{E}_t'\neq\emptyset$ such that

$$I\big(M;X(E_t)\,\big|\,X(\mathscr{E}_t')\big)>0, \tag{99}$$

which, by (96), is equivalent to

$$UI\big(M:X(E_t)\setminus X(\mathscr{E}_t')\big)+SI\big(M:X(E_t);X(\mathscr{E}_t')\big)>0. \tag{100}$$

However, since $I\big(M;X(E_t')\big)=0$, we must have $UI\big(M:X(E_t)\setminus X(\mathscr{E}_t')\big)=0$ by (95) and (97). Hence,

$$\exists\ \mathscr{E}_t'\subseteq\mathscr{E}_t\setminus\{E_t\} \quad \text{s.t.} \quad SI\big(M:X(E_t);X(\mathscr{E}_t')\big)>0. \tag{101}$$

So the implication in the forward direction holds.

($\Leftarrow$) For the converse, suppose that E_t has no M-information flow. That is,

$$I\big(M;X(E_t)\,\big|\,X(\mathscr{E}_t')\big)=0 \quad \forall\ \mathscr{E}_t'\subseteq\mathscr{E}_t\setminus\{\mathscr{E}_t\}. \tag{102}$$

By (96), this implies that

$$\begin{aligned} &UI\big(M:X(E_t)\setminus X(\mathscr{E}_t')\big) \\ &+SI\big(M:X(E_t);X(\mathscr{E}_t')\big)=0 \quad \forall\ \mathscr{E}_t'\subseteq\mathscr{E}_t\setminus\{\mathscr{E}_t\}. \end{aligned} \tag{103}$$

Since UI and SI are both non-negative by (97), we must have that

$$SI\big(M:X(E_t);X(\mathscr{E}_t')\big)=0 \quad \forall\ \mathscr{E}_t'\subseteq\mathscr{E}_t\setminus\{\mathscr{E}_t\}. \tag{104}$$

This proves the converse. ∎

APPENDIX D
MISCELLANEOUS PROOFS FROM SECTION V

A. Proof of Lemma 10

Proof of Lemma 10: Consider a subset of hidden nodes $\mathscr{H}_t'\subseteq\mathscr{H}_t$ that is not M-relevant. Then, by Definition 14, $\mathbb{Q}(\mathscr{H}_t')$ carries no M-information flow in $\mathcal{G}$. This means that

$$\forall\ \mathscr{E}_t'\subseteq\mathscr{E}_t, \quad I\big(M;X(\mathbb{Q}(\mathscr{H}_t'))\,\big|\,X(\mathscr{E}_t')\big)=0. \tag{105}$$

Specifically, taking $\mathscr{E}_t'=\widetilde{\mathscr{E}}_t$, we have

$$I\big(M;X(\mathbb{Q}(\mathscr{H}_t'))\,\big|\,X(\widetilde{\mathscr{E}}_t)\big)=0. \tag{106}$$

Therefore, by Definition 15, $\mathscr{H}_t'$ is M-derived. Thus, if $\mathscr{H}_t'$ is *not* M-relevant, it *is* M-derived. Taking the contrapositive, if $\mathscr{H}_t$ is *not* M-derived, then it *is* M-relevant. ∎

B. Proof of proposition 11

Proof of Proposition 11: We are given that

$$I\big(M;X(\widetilde{\mathscr{E}}_{t+1})\,\big|\,X(\widetilde{\mathscr{E}}_t)\big)>0, \tag{107}$$

and must prove that the hidden nodes at time t, $\mathscr{H}_t$, are *not* M-derived.

First note that, since $\mathbb{Q}(\widetilde{\mathscr{V}}_{t+1})=\widetilde{\mathscr{E}}_{t+1}\cup(\widetilde{\mathscr{V}}_{t+1}\times\mathscr{H}_{t+2})$, we must have

$$\begin{aligned} &I\big(M;X(\mathbb{Q}(\widetilde{\mathscr{V}}_{t+1}))\,\big|\,X(\widetilde{\mathscr{E}}_t)\big) \\ &\quad= I\big(M;X(\widetilde{\mathscr{E}}_{t+1}),X(\widetilde{\mathscr{V}}_{t+1}\times\mathscr{H}_{t+2})\,\big|\,X(\widetilde{\mathscr{E}}_t)\big) && (108)\\ &\quad= I\big(M;X(\widetilde{\mathscr{E}}_{t+1})\,\big|\,X(\widetilde{\mathscr{E}}_t)\big) \\ &\qquad+ I\big(M;X(\widetilde{\mathscr{V}}_{t+1}\times\mathscr{H}_{t+2})\,\big|\,X(\widetilde{\mathscr{E}}_{t+1}),X(\widetilde{\mathscr{E}}_t)\big) && (109)\\ &\quad\geq I\big(M;X(\widetilde{\mathscr{E}}_{t+1})\,\big|\,X(\widetilde{\mathscr{E}}_t)\big) && (110)\\ &\quad> 0, && (111) \end{aligned}$$

where the last line follows from the fact that conditional mutual information is non-negative, and from (107).

Next, observe that Local Markovity conditions (Proposition 6) *must* hold on the *entire* graph $\mathcal{G}$, which consists of both observed and hidden nodes. If we apply the Local Markovity condition to $\widetilde{\mathscr{V}}_{t+1}$, we have M—$X(\mathcal{P}(\widetilde{\mathscr{V}}_{t+1}))$—$X(\mathbb{Q}(\widetilde{\mathscr{V}}_{t+1}))$, or in other words

$$I\big(M;X(\mathbb{Q}(\widetilde{\mathscr{V}}_{t+1}))\,\big|\,X(\mathcal{P}(\widetilde{\mathscr{V}}_{t+1}))\big)=0. \tag{112}$$

Note that $\mathcal{P}(\widetilde{\mathscr{V}}_{t+1})=\widetilde{\mathscr{E}}_t\cup\widetilde{\mathbb{Q}}(\mathscr{H}_t)$, where $\widetilde{\mathbb{Q}}(\mathscr{H}_t):=\mathscr{H}_t\times\widetilde{\mathscr{V}}_{t+1}$ is the subset comprising outgoing edges of $\mathscr{H}_t$ that go to $\widetilde{\mathscr{V}}_{t+1}$. Therefore,

$$I\big(M;X(\mathbb{Q}(\widetilde{\mathscr{V}}_{t+1}))\,\big|\,X(\widetilde{\mathscr{E}}_t),X(\widetilde{\mathbb{Q}}(\mathscr{H}_t))\big)=0. \tag{113}$$

Expanding this conditional mutual information, we get

$$\begin{aligned} &I\big(M;X(\mathbb{Q}(\widetilde{\mathscr{V}}_{t+1})),X(\widetilde{\mathbb{Q}}(\mathscr{H}_t))\,\big|\,X(\widetilde{\mathscr{E}}_t)\big) \\ &\qquad- I\big(M;X(\widetilde{\mathbb{Q}}(\mathscr{H}_t))\,\big|\,X(\widetilde{\mathscr{E}}_t)\big)=0. \end{aligned} \tag{114}$$

So we have

$$\begin{aligned} &I\big(M;X(\widetilde{\mathbb{Q}}(\mathscr{H}_t))\,\big|\,X(\widetilde{\mathscr{E}}_t)\big) \\ &\quad= I\big(M;X(\mathbb{Q}(\widetilde{\mathscr{V}}_{t+1})),X(\widetilde{\mathbb{Q}}(\mathscr{H}_t))\,\big|\,X(\widetilde{\mathscr{E}}_t)\big) && (115)\\ &\quad= I\big(M;X(\mathbb{Q}(\widetilde{\mathscr{V}}_{t+1}))\,\big|\,X(\widetilde{\mathscr{E}}_t)\big) \\ &\qquad+ I\big(M;X(\widetilde{\mathbb{Q}}(\mathscr{H}_t))\,\big|\,X(\mathbb{Q}(\widetilde{\mathscr{V}}_{t+1})),X(\widetilde{\mathscr{E}}_t)\big) && (116)\\ &\quad> 0, \end{aligned}$$

where the final inequality follows from (111) and the fact that conditional mutual information is non-negative. Finally, since $\widetilde{\mathbb{Q}}(\mathscr{H}_t) \subset \mathbb{Q}(\mathscr{H}_t)$, we have that $I\big(M; X(\mathbb{Q}(\mathscr{H}_t)) \,\big|\, X(\widetilde{\mathscr{E}}_t)\big) > 0$, just as we showed in equations (108)–(111). Hence, the Markov chain M—$X(\widetilde{\mathscr{E}}_t)$—$X(\mathbb{Q}(\mathscr{H}_t))$ does not hold, so by Definition 15, $\mathscr{H}_t$ are not M-derived. ∎

APPENDIX E
ON THE UNIQUENESS OF OUR DEFINITION OF INFORMATION FLOW

From the perspective of designing an axiomatic framework, it is desirable to find a minimal set of properties that gives rise to a unique definition of information flow. Although Property 1 helped us motivate a definition for information flow, it did not uniquely specify a definition. Indeed, the all-zero function as well as the all-ones function also satisfy the property, although they are not particularly useful definitions of information flow.

In this section, we provide a set of properties that uniquely leads to our definition of information flow. However, we must acknowledge that we arrived at these properties with the benefit of hindsight, after having proved many other properties of our definition. As such, they are mathematically very similar to our definition, and one might feel uncomfortable with the idea of imposing such a set of properties at the very outset. Our goal here is only to begin a discussion in this direction: a search for a more abstract set of properties that leads to a unique definition of information flow would be a worthy endeavour in future.

Property 4: Let $\mathscr{C}$ be a computational system, and let $\mathscr{F}_M : \mathscr{C} \to \{0, 1\}$ be an indicator of the presence of information flow about M on an edge. That is, $\mathscr{F}_M(E) = 1$, if information about M flows on the edge $E \in \mathscr{C}$, and $\mathscr{F}_M(E) = 0$ otherwise. We now state three conditions $\mathscr{F}_M$ must satisfy, which naturally leads to our definition of information flow (Definition 4):

4a) $\mathscr{F}_M(E_t) = 1$ if $I\big(M; X(E_t)\big) > 0$

4b) $\mathscr{F}_M(E_t) = 1$ if $\exists\ \mathscr{E}'_t \subseteq \mathscr{E}_t \backslash \{E_t\}$ s.t.
$$I\big(M; X(\mathscr{E}'_t) \,|\, X(E_t)\big) > I\big(M; X(\mathscr{E}'_t)\big)$$

4c) $\mathscr{F}_M(E_t) = 0$ if $I\big(M; X(E_t) \,|\, X(\mathscr{E}'_t)\big) = 0\ \forall\ \mathscr{E}'_t \subseteq \mathscr{E}_t$.

Property 4a is a very natural and intuitive requirement for information flow. Property 4b states that an edge should be considered to carry information about M, if upon conditioning, its transmission *increases* the information that some set $X(\mathscr{E}'_t)$ conveys about M. Property 4c is reminiscent of the separability property from Proposition 9, and states that if an edge has no dependence with M, no matter what other transmission is conditioned upon, then it can carry no information flow about M.

Effectively, Property 4a states that if an edge has unique or redundant information about M, then it must carry information flow, while Property 4b states that if an edge has synergistic information about M along with some other set of transmissions, then it must carry information flow. Finally, Property 4c states that if all three of these components are absent, then that edge carries no information flow. This also explains how, if any one of these three properties is absent, our definition is no longer unique.

As we acknowledged previously, some of these properties could be seen as too restrictive or contrived, and a more abstract set of properties is certainly desirable. Nevertheless, these properties do uniquely identify our definition of information flow.

Proposition 12 (Uniqueness): If $\mathscr{F}_M$ is an indicator of information flow that satisfies the conditions in Property 4, then $\mathscr{F}_M(E_t) = 1$ if and only if E_t has M-information flow, per Definition 4.

Proof: ($\Rightarrow$) Suppose the edge E_t has no M-information flow per Definition 4. This directly implies the condition in Property 4c. Hence, $\mathscr{F}_M(E_t) = 0$. This proves that if $\mathscr{F}_M(E_t) = 1$, the edge E_t must have M-information flow.

($\Leftarrow$) Suppose the edge E_t *has* M-information flow per Definition 4. Then,

$$\exists\ \mathscr{E}'_t \subseteq \mathscr{E}_t \backslash \{E_t\} \quad \text{s.t} \quad I\big(M; X(E_t) \,\big|\, X(\mathscr{E}'_t)\big) > 0. \tag{117}$$

If $\mathscr{E}'_t = \emptyset$, $I\big(M; X(E_t)\big) > 0$, so by Property 4a, $\mathscr{F}_M(E_t) = 1$. If $I\big(M; X(E_t)\big) = 0$, then (117) guarantees the existence of some $\mathscr{E}'_t \neq \emptyset$ such that

$$I\big(M; X(E_t) \,\big|\, X(\mathscr{E}'_t)\big) > 0 \tag{118}$$

$$I\big(M; X(\mathscr{E}'_t)\big) + I\big(M; X(E_t) \,\big|\, X(\mathscr{E}'_t)\big) \overset{(a)}{>} I\big(M; X(\mathscr{E}'_t)\big) \tag{119}$$

$$I\big(M; X(E_t), X(\mathscr{E}'_t)\big) \overset{(b)}{>} I\big(M; X(\mathscr{E}'_t)\big) \tag{120}$$

$$I\big(M; X(E_t)\big) + I\big(M; X(\mathscr{E}'_t) \,\big|\, X(E_t)\big) \overset{(c)}{>} I\big(M; X(\mathscr{E}'_t)\big) \tag{121}$$

$$I\big(M; X(\mathscr{E}'_t) \,\big|\, X(E_t)\big) \overset{(d)}{>} I\big(M; X(\mathscr{E}'_t)\big), \tag{122}$$

where in (a), we simply added $I\big(M; X(\mathscr{E}'_t)\big)$ to both sides; in (b) and (c), we used the chain rule in two different ways; and in (d), we used the fact that $I\big(M; X(E_t)\big) = 0$. So, by Property 4b, we have that $\mathscr{F}_M(E_t) = 1$. This proves the converse. ∎

Remark: It should be noted that Definition 4 only specifies *whether or not* a given edge has M-information flow. It does not *quantify* this flow. So Proposition 12 demonstrates the uniqueness of our definition up to an unspecified information volume. If we require that the conditions in Property 4 hold, then any quantitative definition of information flow will go to zero at an edge if and only if the M-information flow carried by that edge is zero.

APPENDIX F
MISCELLANEOUS DERIVATIONS FROM SECTION VI

A. Derivation of Expressions in the Second FFT Example from Section VI-B

Here, we derive the expressions used in Figure 13. Recall that $Y_i = \omega^{-iM}/4$, where $\omega = e^{-j2\pi/4} = -j$.

$$\begin{aligned} Y_{02} &= Y_0 + Y_2 \\ &= \frac{1}{4} + \frac{\omega^{-2M}}{4} = \frac{1}{4}(1 + \omega^{-2M}) \end{aligned} \tag{123}$$

$$Y_{13} = Y_1 + Y_3 = \frac{\omega^{-M}}{4} + \frac{\omega^{-3M}}{4} = \frac{\omega^{-M}}{4}(1+\omega^{-2M}) \tag{124}$$

$$Y'_{02} = Y_0 + \omega^2 Y_2 = \frac{1}{4} + (-1)\frac{\omega^{-2M}}{4} = \frac{1}{4}(1-\omega^{-2M}) \tag{125}$$

$$Y'_{13} = Y_1 + \omega^2 Y_3 = \frac{\omega^{-M}}{4} + (-1)\frac{\omega^{-3M}}{4} = \frac{\omega^{-M}}{4}(1-\omega^{-2M}) \tag{126}$$

Next, we show that these intermediate values actually yield the expected values of $\widetilde{Y}$.

$$\widetilde{Y}_0 = Y_{02} + Y_{13} = \frac{1}{4}(1+\omega^{-2M}+\omega^{-M}+\omega^{-3M}) \tag{127}$$

$$= \begin{cases} \frac{1}{4}(1+1+1+1), & M=0 \\ \frac{1}{4}(1+j+j^2+j^3), & M=1 \end{cases} \tag{128}$$

$$= 1 - M \tag{129}$$

$$\widetilde{Y}_1 = Y'_{02} + \omega Y'_{13} = \frac{1}{4}(1-\omega^{-2M}+\omega^{1-M}-\omega^{1-3M}) \tag{130}$$

$$= \begin{cases} \frac{1}{4}(1-1+\omega-\omega), & M=0 \\ \frac{1}{4}(1-j^2+1-j^2), & M=1 \end{cases} \tag{131}$$

$$= M \tag{132}$$

$$\widetilde{Y}_2 = Y_{02} + \omega^2 Y_{13} = \frac{1}{4}(1+\omega^{-2M}+\omega^{2-M}+\omega^{2-3M}) \tag{133}$$

$$= \frac{1}{4}(1+\omega^{-2M}-\omega^{-M}-\omega^{-3M}) \tag{134}$$

$$= \begin{cases} \frac{1}{4}(1+1-1-1), & M=0 \\ \frac{1}{4}(1-1-\omega^{-1}+\omega^{-1}), & M=1 \end{cases} \tag{135}$$

$$= 0 \tag{136}$$

$$\widetilde{Y}_3 = Y'_{02} + \omega^3 Y'_{13} = \frac{1}{4}(1-\omega^{-2M}+\omega^{3-M}-\omega^{3-3M}) \tag{137}$$

$$= \frac{1}{4}(1-\omega^{-2M}+\omega^3(\omega^{-M}-\omega^{-3M})) \tag{138}$$

$$= \begin{cases} \frac{1}{4}(1-1-\omega(1-1)), & M=0 \\ \frac{1}{4}(1-(-1)-\omega(\omega^{-1}+\omega^{-1})), & M=1 \end{cases} \tag{139}$$

$$= 0 \tag{140}$$

B. Derivation of the Markov Chain Failure in Section VI-C

We wish to show that in the canonical example from Section VI-C, M—$[\widehat{M}_1,\ \widehat{M}_2]$—$(M-\widehat{M}_2)$ is not a valid Markov chain. Recall that $Z_1, Z_2, Z_3 \sim$ i.i.d. $\mathcal{N}(0,\sigma^2)$ and $M \sim \mathcal{N}(0,1)$. Let $h(\cdot)$ denote differential entropy. Then,

$$I(M; M-\widehat{M}_2, \widehat{M}_2) \geq I(M;\widehat{M}_3) = h(\widehat{M}_3) - h(\widehat{M}_3 \,|\, M) \tag{141}$$

$$= \frac{1}{2}\log\Big(2\pi e\Big(1+\frac{\sigma^2}{3}\Big)\Big) - \frac{1}{2}\log\Big(2\pi e\Big(\frac{\sigma^2}{3}\Big)\Big) \tag{142}$$

$$= \frac{1}{2}\log\Big(1+\frac{3}{\sigma^2}\Big). \tag{143}$$

Here, we started with the Data Processing Inequality, and then used the fact that if $Y \sim \mathcal{N}(0,\sigma^2)$ is a zero-mean scalar Gaussian random variable with variance σ^2, then its differential entropy is given by [73, Thm. 8.4.1]

$$h(Y) = \frac{1}{2}\log(2\pi e\sigma^2) \text{ nats.} \tag{144}$$

Next, note that since $\widehat{M}_1 = M + Z_1$ and $\widehat{M}_2 = M + \frac{1}{2}(Z_1+Z_2)$, $\widehat{M}_1$ has no extra information about M, given $\widehat{M}_2$. This is obvious when we think of $\widehat{M}_1$ as being $\widehat{M}_1 = \widehat{M}_2 + Z'$, where $Z' = \frac{1}{2}(Z_1 - Z_2)$, and it can be shown that $Z' \perp\!\!\!\perp \widehat{M}_2$:

$$\mathbb{E}[\widehat{M}_2 Z'] = \mathbb{E}\Big[\Big(M+\frac{1}{2}(Z_1+Z_2)\Big)Z'\Big] \tag{145}$$

$$= \mathbb{E}[MZ'] + \frac{1}{4}\mathbb{E}\big[(Z_1+Z_2)(Z_1-Z_2)\big] \tag{146}$$

$$= 0 + \frac{1}{4}\mathbb{E}[Z_1^2 - Z_2^2] \tag{147}$$

$$= \frac{1}{4}(\sigma^2-\sigma^2) = 0. \tag{148}$$

Since all variables involved are zero-mean Gaussians, this naturally implies that $\widehat{M}_2 \perp\!\!\!\perp Z'$. Thus, from our previous argument, $\widehat{M}_1$ has no extra information about M when given $\widehat{M}_2$, or in other words, M—$\widehat{M}_2$—$\widehat{M}_1$ is a valid Markov chain. Therefore,

$$I(M;\widehat{M}_1,\widehat{M}_2) = I(M;\widehat{M}_2) + I(M;\widehat{M}_1 \,|\, \widehat{M}_2) \tag{149}$$

$$= I(M;\widehat{M}_2) + 0 \tag{150}$$

$$= \frac{1}{2}\log\Big(1+\frac{2}{\sigma^2}\Big), \tag{151}$$

derived in the same way as (143). From (143) and (151), we can conclude that $I(M;\widehat{M}_3) > I(M;\widehat{M}_2)$, and therefore

$$I(M; M-\widehat{M}_2, \widehat{M}_2) > I(M;\widehat{M}_1,\widehat{M}_2) \tag{152}$$

$$\Rightarrow\ I(M; M-\widehat{M}_2, \widehat{M}_2, \widehat{M}_1) > I(M;\widehat{M}_1,\widehat{M}_2) \tag{153}$$

$$\Rightarrow\ I(M; M-\widehat{M}_2, \widehat{M}_2, \widehat{M}_1) - I(M;\widehat{M}_1,\widehat{M}_2) > 0 \tag{154}$$

$$\Rightarrow\ I(M; M-\widehat{M}_2 \,|\, \widehat{M}_1,\widehat{M}_2) > 0. \tag{155}$$

Thus, the stated Markov chain, M—$[\widehat{M}_1,\ \widehat{M}_2]$—$(M-\widehat{M}_2)$, cannot hold.

Praveen Venkatesh [S'15] received the B.Tech. (honors) degree in Electrical Engineering from the Indian Institute of Technology, Madras, India, in 2014. Currently, he is a Ph.D. candidate in the Electrical and Computer Engineering Department, Carnegie Mellon University, Pittsburgh, PA, USA.

His interests lie broadly in the intersection of information theory, signal processing, neuroscience and neuroengineering. Over the last few years, he has worked on understanding information flow in the brain, on algorithms for source localization, on understanding the fundamental limits of EEG source localization, and on experimental design and data analysis with EEG and ECoG data.

Mr. Venkatesh is a recipient of the Henry L. Hillman Presidential Fellowship, the Dowd Fellowship from the College of Engineering, and the CMLH Fellowship in Digital Health from the Center for Machine Learning and Health at Carnegie Mellon University.

Sanghamitra Dutta [S'15] received her B.Tech in electronics and electrical communication engineering from the Indian Institute of Technology, Kharagpur, India, in 2015. She is a doctoral candidate in the Department of Electrical and Computer Engineering at Carnegie Mellon University, PA, USA.

Her research interests revolve around machine learning and information theory. She is currently focussed on addressing the emerging trust issues in machine learning concerning fairness, privacy and reliability. In one of her works, she has proposed a rigorous quantification of bias in machine learning with respect to protected attributes, such as gender, race, etc. that enables one to check if the bias arose purely due to business necessities (drawing inspiration from the business necessity defense in the disparate impact law). Her work bridges the fields of information theory, causality, reliability and machine learning. In her prior work, she has also examined problems in reliable computing, proposing novel algorithmic solutions for large-scale machine-learning in the presence of faults and failures, using tools from coding theory (an emerging area called "coded computing"). Her results on coded computing address problems that have been open for several decades and have received substantial attention from across communities.

Ms. Dutta is a recipient of the 2019 K&L Gates Presidential Fellowship, 2019 Axel Berny Presidential Graduate Fellowship, 2017 Tan Endowed Graduate Fellowship, 2016 Prabhu and Poonam Goel Graduate Fellowship, and the 2014 HONDA Young Engineer and Scientist Award.

Pulkit Grover (Ph.D. UC Berkeley'10, B.Tech, M.Tech IIT Kanpur) is an associate professor at CMU (2013-present). His main contributions to science are towards developing and experimentally validating a new theory of information (fundamental limits, practical designs) for efficient and reliable communication, computing, sensing, and control, e.g. by incorporating novel circuit-energy models and developing new mathematical tools for information flow analyses. To apply these ideas to a variety of problems including novel biomedical systems, his lab works extensively with system and device engineers, neuroscientists, and doctors. Specifically, work of his foundations of neuroengineering lab is focused on tools (theoretical, computational, and hardware) for understanding, diagnosing, and treating disorders such as epilepsy, Parkinson's, and traumatic brain injuries.

Pulkit received the 2010 best student paper award at IEEE Conference on Decision and Control; the 2010 IEEE ISIT best paper finalist nomination; the 2011 Eli Jury Dissertation Award from UC Berkeley; the 2012 Leonard G. Abraham best journal paper award from the IEEE ComSoc; a 2014 NSF CAREER award; a 2015 Google Research Award; a 2018 CMU Dean's Early Career Fellowship; a 2018 inaugural award from the Chuck Noll Foundation for Brain Injury Research; the 2018 Joel and Ruth Spira Excellence in Teaching Award from CMU, and the 2019 best tutorial paper award from IEEE ComSoc. He's learning how to play the sax and enjoys his free time with his wife, Kristen, and son, Utsah.