Journal
Journal of Statistical Physics, 161(2)

ISSN
0022-4715

Authors
Barnett, N
Crutchfield, JP

Publication Date
2015-10-30

DOI
10.1007/s10955-015-1327-5

Peer reviewed

J Stat Phys (2015) 161:404–451
DOI 10.1007/s10955-015-1327-5

Computational Mechanics of Input–Output Processes: Structured Transformations and the ϵ-Transducer

Nix Barnett[1,2] · **James P. Crutchfield**[1,2,3]

Received: 8 December 2014 / Accepted: 9 July 2015 / Published online: 12 August 2015

Abstract Computational mechanics quantifies structure in a stochastic process via its causal states, leading to the process's minimal, optimal predictor—the ϵ-machine. We extend computational mechanics to communication channels coupling two processes, obtaining an analogous optimal model—the ϵ-transducer—of the stochastic mapping between them. Here, we lay the foundation of a structural analysis of communication channels, treating joint processes and processes with input. The result is a principled structural analysis of mechanisms that support information flow between processes. It is the first in a series on the structural information theory of memoryful channels, channel composition, and allied conditional information measures.

Keywords Sequential machine · Communication channel · Finite-state transducer · Statistical complexity · Causal state · Minimality · Optimal prediction · Subshift endomorphism

1 Introduction

Arguably, the distinctive character of natural and engineered systems lies in their organization. This is in contrast to differences, say, in how random they are or in their temperature.

✉ James P. Crutchfield
chaos@ucdavis.edu
http://csc.ucdavis.edu/~chaos/

Nix Barnett
nix@math.ucdavis.edu
http://math.ucdavis.edu/~nix/

1 Complexity Sciences Center, University of California at Davis, One Shields Avenue, Davis, CA 95616, USA

2 Mathematics Department, University of California at Davis, One Shields Avenue, Davis, CA 95616, USA

3 Physics Department, University of California at Davis, One Shields Avenue, Davis, CA 95616, USA

Computational mechanics provides an analytical and constructive way to determine a system's organization [1], supplementing the well developed statistical physics view of systems in terms of disorder—via thermodynamic entropy and free energy. The contrast begs a question, though, how does organization arise?

As a step to an answer, we extend computational mechanics from describing individual systems to analyze organization in transformations between systems. We build on its well established methods to describe a calculus for detecting the emergence of structure. Indeed, natural systems are nothing, if not the result of transformations of energy, information, or both. There is no lack of examples: Filtering measurement time series to discover the temporal behavior of hidden states [2]; Maxwell's Demon translating measurement information to control actions that rectify thermal fluctuations into work, locally defeating Thermodynamics' Second Law [3–5]; sensory transduction in the retina that converts light intensity and spectra into neural spike trains [6]; perception-action cycles in which an agent decides its future behavior based on its interpretation of its environment's state [7,8]; and, finally, firms that transform raw materials into finished goods [9].

We refer to our objects of study as *structured transformations* to emphasize the particular focus on basic questions of organization. How complex is a transformation? What and how many resources are required to implement it? What does it add to an input in producing an output? Randomness, structure, or both? What is thrown away? How is its own organization reflected in that of its output?

The framework addresses these questions, both quantitatively and from first principles. It is the first in a series. Foremost, it's burden is to lay the groundwork necessary to answer these questions. Sequels introduce information measures to classify the kinds of information processing in joint input–output processes and in structured transformations. To delineate the underlying mechanisms that produce organization, they analyze a range of examples and describe a number of interesting, even counterintuitive, properties of structured transformations.

The questions posed are basic, so there is a wide range of applications and of historical precedents. Due to this diversity and to avoid distracting from the development, we defer reviewing them and related work until later, once the context has been set and the focus, benefits, and limitations of our approach are clear.

The following analyzes communication channels and channel composition in terms of intrinsic computation [1,10]. As such, it and the entire series, for that matter, assume familiarity with stochastic processes at the level of Ref. [11], information theory at the level of Refs. [12,13], and computational mechanics at the level of Refs. [14,15]. These serve as the default sources for statements in our review.

We first present a brief overview and relevant notation for how computational mechanics describes processes. We extend this to describe controlled processes—processes with input. Several examples that illustrate the basic kinds of input–output process are then given, by way of outlining an organizational classification scheme. At that point, we describe the global ϵ-machine for joint input–output processes. Using this we introduce the ϵ-transducer, defining its structural complexity and establishing its optimalities. We close with a thorough analysis of the example input–output processes and compare and contrast ϵ-transducers with prior work.

2 Processes and Their Presentations

2.1 Stationary, Ergodic Processes

The temporal *stochastic process* we consider is a one-dimensional chain $\ldots Y_{-2} Y_{-1} Y_0 Y_1 Y_2 \ldots$ of discrete random variables $\{Y_t\}_{t\in\mathbb{Z}}$ that take values $\{y_t\}_{t\in\mathbb{Z}}$ over a finite or countable alphabet $\mathcal{Y}$. A finite block $Y_i Y_{i+1} \ldots Y_{j-1}$ of variables with $t \in [i, j)$ is denoted $Y_{i:j}$. The left index is always inclusive, the right always exclusive. Let $\overleftrightarrow{Y}$ denote the bi-infinite chain $Y_{-\infty:\infty}$, and let $\overleftrightarrow{\mathcal{Y}}$ denote the set of all bi-infinite sequences $\overleftrightarrow{y}$ with alphabet $\mathcal{Y}$. $\overleftarrow{Y_t} = \ldots Y_{t-2} Y_{t-1}$ is the *past* leading up to time t, not including t, and $\overrightarrow{Y_t} = Y_t Y_{t+1} \ldots$ is the *future* leading from it, including t.

We can also use the time index notation above to specify the time origin of a process or bi-infinite sequence. This is useful when we are comparing two such processes or sequences. For example, if we wish to say that $\overleftrightarrow{Y}$ is $\overleftrightarrow{X}$ delayed by three time steps, this can be written as $\overleftrightarrow{Y_3} = \overleftrightarrow{X_0}$. To indicate the time origin in *specific* realized symbol values in a chain, we place a period before the symbol occurring at $t = 0$: e.g., $\overleftrightarrow{y} = \ldots acbb.caba \ldots$, where $y_0 = c$. Finally, if the word $abcd$ follows a sequence of random variables $Y_{i:j}$ we denote this by $Y_{i:j+4} = Y_{i:j}abcd$, for example. A word occurring before a sequence of random variables is denoted by a similar concatenation rule; e.g., $Y_{i-4:j} = abcdY_{i:j}$.

A stochastic process is defined by its *word distributions*:

$$\mathbb{P}(Y_{t:t+L}) \equiv \{\mathbb{P}(Y_{t:t+L} = y_{t:t+L})\}_{y_{t:t+L}\in\mathcal{Y}^L}, \tag{1}$$

for all $L \in \mathbb{Z}^+$ and $t \in \mathbb{Z}$.

We will often use an equivalent definition of a stochastic process as a random variable defined over the set of bi-infinite sequences $\overleftrightarrow{\mathcal{Y}}$. In this case, a stochastic process is defined by an indexed set of probabilities of the form:

$$\mathbb{P}(\overleftrightarrow{Y}) \equiv \{\mathbb{P}(\overleftrightarrow{Y} \in \sigma)\}_{\sigma\subseteq\overleftrightarrow{\mathcal{Y}}}, \tag{2}$$

where σ is a measurable set of bi-infinite sequences. We can obtain a process' word probabilities by selecting appropriate measurable subsets—*cylinder sets*—that correspond to holding the values of a contiguous subset of random variables fixed.

In the following, we consider only *stationary processes*—those invariant under time translation:

$$\mathbb{P}(Y_{t:t+L}) = \mathbb{P}(Y_{0:L}) \text{ and}$$
$$\mathbb{P}(\overleftrightarrow{Y_t}) = \mathbb{P}(\overleftrightarrow{Y_0}),$$

for all $t \in \mathbb{Z}$ and $L \in \mathbb{Z}^+$. This property ensures that a process's behavior has no *explicit* dependence on time origin.

We will also primarily limit the discussion to *ergodic* stationary processes—processes where any realization $\overleftrightarrow{y}$ gives good empirical estimates $\widehat{\mathbb{P}}(Y_{0:L})$ of the process's true word probabilities $\mathbb{P}(Y_{0:L})$ [16]. That is, for any finite realization $y_{0:M}$, the empirical estimate $\widehat{\mathbb{P}}(w)$ of a word $w = w_0 w_1 \ldots w_{L-1}$ of length L, converges almost surely to the true process probability $\mathbb{P}(Y_{0:L} = w)$ as $M \to \infty$, where:

$$\widehat{\mathbb{P}}(w) \equiv \sum_{t=0}^{M-L} \frac{\mathbf{I}_w(y_{t:t+L})}{M-L+1},$$

and the indicator function $\mathbf{I}_w(y_{t:t+L})$ equals 1 when $y_{t:t+L} = w$, and 0 otherwise. This property ensures, among other things, that any particular realization of the process reflects the behavior of the process in general—a useful property when we do not have freedom to re-initialize the system at will.

2.2 Examples

To illustrate key ideas in the development, we use several example processes, all with the binary alphabet $\mathcal{Y} = \{0, 1\}$. They are used repeatedly in the following and in the sequels.

2.2.1 Biased Coin Process

The *Biased Coin Process* is an independent, identically distributed (IID) process, where word probabilities factor into a product of single-variable probabilities:

$$\mathbb{P}(Y_{0:L}) = \mathbb{P}(Y_0)\mathbb{P}(Y_1)\cdots\mathbb{P}(Y_{L-1}),$$

where $\mathbb{P}(Y_t) = \mathbb{P}(Y_0)$ for all t. If n is the number of 0s in $y_{0:L}$, then $\mathbb{P}(y_{0:L}) = p^n(1-p)^{L-n}$, where $p = \mathbb{P}(Y_0 = 0)$.

2.2.2 Period-2 Process

The *Period-*2 *Process* endlessly repeats the word 01, starting with either a 0 or a 1 with equal probability. It is specified by the word distributions:

$$\mathbb{P}(Y_{i:j} = (\ldots 1010.1010\ldots)_{i:j}) = \tfrac{1}{2} \text{ and}$$
$$\mathbb{P}(Y_{i:j} = (\ldots 0101.0101\ldots)_{i:j}) = \tfrac{1}{2},$$

where $i, j \in \mathbb{Z}, i < j$.

2.2.3 Golden Mean Process

The *Golden Mean Process* generates all binary sequences, except those with consecutive 0s. After a 1 is generated, the next 0 or 1 appears with equal likelihood. Its word distributions are determined by a Markov chain with states $Y_0 = 0$ and $Y_0 = 1$ having probabilities $\mathbb{P}(Y_0 = 0) = 1/3$ and $\mathbb{P}(Y_0 = 1) = 2/3$, respectively, and transition probabilities:

$$\begin{aligned}
\mathbb{P}(Y_1 = 0|Y_0 = 0) &= 0,\\
\mathbb{P}(Y_1 = 1|Y_0 = 0) &= 1,\\
\mathbb{P}(Y_1 = 0|Y_0 = 1) &= \tfrac{1}{2}, \text{ and}\\
\mathbb{P}(Y_1 = 1|Y_0 = 1) &= \tfrac{1}{2}.
\end{aligned} \tag{3}$$

2.2.4 Even Process

The *Even Process* generates all binary sequences, except that a 1 always appears in an even-length block of 1s bordered by 0s. After an even number of 1s are generated, the next 0 or 1 appears with equal likelihood. Notably, the Even Process cannot be represented by a Markov chain of any finite order or, equivalently, by any finite list of conditional probabilities over words. As we will see, though, it can be represented concisely by a finite-state hidden Markov model. To see this, we must first introduce alternative models—here called *presentations*—for a given process.

2.3 Optimal Presentations

We can completely, and rather prosaically, specify a stationary process by listing its set of word probabilities, as in Eq. (1), or conditional probabilities, as in Eq. (3). As with the Even Process, however, generally these sets are infinite and so one prefers to use finite or, at least, more compact descriptions. Fortunately, there is a canonical *presentation* for any stationary process—the ϵ-machine of computational mechanics [10,14]—that is its unique, optimal, unifilar generator of minimal size; which we now define.

Given a process's word distribution, its ϵ-machine is constructed by regarding any two infinite pasts $\overleftarrow{y}$ and $\overleftarrow{y}'$ as equivalent when they lead to the same distribution over infinite futures—the same *future morphs*, of the form $\mathbb{P}(\overrightarrow{Y}\,|\,\overleftarrow{y})$. This grouping is given by the *causal equivalence relation* $\sim_\epsilon$:

$$\overleftarrow{y} \sim_\epsilon \overleftarrow{y}' \iff \mathbb{P}(\overrightarrow{Y}\,|\,\overleftarrow{Y} = \overleftarrow{y}) = \mathbb{P}(\overrightarrow{Y}\,|\,\overleftarrow{Y} = \overleftarrow{y}'). \tag{4}$$

The equivalence classes of $\sim_\epsilon$ partition the set $\overleftarrow{\boldsymbol{\mathcal{Y}}}$ of all allowed pasts. The classes are the process's *causal states*. The indexed set of causal states is denoted by $\boldsymbol{\mathcal{S}}$ and has elements σ_i. Note that i is not a time index, but an element of an index set with the same cardinality as $\boldsymbol{\mathcal{S}}$. The associated random variable over alphabet $\boldsymbol{\mathcal{S}}$ is denoted by S. The ϵ*-map* is a function $\epsilon : \overleftarrow{\boldsymbol{\mathcal{Y}}} \to \boldsymbol{\mathcal{S}}$ that takes a given past to its corresponding causal state or, equivalently, to the set of pasts to which it is causally equivalent:

$$\begin{aligned} \epsilon(\overleftarrow{y}) &= \sigma_i \\ &= \{\overleftarrow{y}' : \overleftarrow{y} \sim_\epsilon \overleftarrow{y}'\}. \end{aligned}$$

The ϵ-map induces a dynamic over pasts: Appending a new symbol y_t to past $\overleftarrow{y_t}$ produces a new past $\overleftarrow{y}_{t+1} = \overleftarrow{y_t} y_t$. This, in turn, defines a stochastic process—the *causal-state process*—with random variable chain $\overleftrightarrow{S} = S_{-\infty:\infty} = \ldots S_{-1} S_0 S_1 \ldots$, where at time t each S_t takes on some value $s_t = \sigma_i \in \boldsymbol{\mathcal{S}}$. The relationship between a process's pasts and its causal states is summarized in Fig. 1. The map from the observed chain $\overleftrightarrow{Y}$ to the internal state chain $\overleftrightarrow{S}$ is the *causal state filter*. (We return to this mapping later on.) The dynamic over causal states is specified by an indexed set $\boldsymbol{\mathcal{T}}$ of *symbol-labeled transition matrices*:

$$\boldsymbol{\mathcal{T}} \equiv \{T^{(y)}\}_{y \in \boldsymbol{\mathcal{Y}}},$$

where $T^{(y)}$ has elements:

$$T^{(y)}_{ij} = \mathbb{P}(S_1 = \sigma_j, Y_0 = y | S_0 = \sigma_i).$$

$$\begin{tikzcd}
\cdots \arrow[r, "y_{t-2}"] & \overleftarrow{y}_{t-1} \arrow[r, "y_{t-1}"] \arrow[d, "\epsilon"'] & \overleftarrow{y}_{t} \arrow[r, "y_{t}"] \arrow[d, "\epsilon"'] & \overleftarrow{y}_{t+1} \arrow[r, "y_{t+1}"] \arrow[d, "\epsilon"'] & \cdots \\
\cdots \arrow[r, "y_{t-2}"] & s_{t-1} \arrow[r, "y_{t-1}"] & s_{t} \arrow[r, "y_{t}"] & s_{t+1} \arrow[r, "y_{t+1}"] & \cdots
\end{tikzcd}$$

Fig. 1 Process lattice: the dynamic inherited by the causal states $s_t = \sigma_i \in \boldsymbol{\mathcal{S}}$ from a process's pasts via the ϵ-map: $s_t = \epsilon(\ldots y_{t-2} y_{t-1}) \xrightarrow{y_t} s_{t+1} = \epsilon(\ldots y_{t-2} y_{t-1} y_t)$

The ϵ-map also induces a distribution π over causal states, with elements:

$$\begin{aligned}\pi(i) &= \mathbb{P}(S_0 = \sigma_i) \\ &= \mathbb{P}\big(\epsilon\big(\overleftarrow{Y}\big) = \sigma_i\big).\end{aligned}$$

Since the process is stationary and the ϵ-map is time independent, π is also stationary. We therefore refer to π as the process's *stationary distribution*. Note that for ergodic processes, the stationary distribution can be calculated directly from the transition matrices alone [17].

The tuple $(\boldsymbol{\mathcal{Y}}, \boldsymbol{\mathcal{S}}, \boldsymbol{\mathcal{T}})$ consisting of the process' alphabet, causal state set, and transition matrix set is the process' ϵ-machine.

The ϵ-machine is a process's unique, maximally predictive, minimal-size unifilar presentation [10,18,19]. In other words, the causal states are as predictive as any rival partition $\boldsymbol{\mathcal{R}}$ of the pasts $\overleftarrow{\boldsymbol{\mathcal{Y}}}$. In particular, any given causal state σ_i is as predictive as any of its pasts $\overleftarrow{y} \in \epsilon^{-1}(\sigma_i)$. Measuring predictive ability via the mutual information between states and future observations, this translates to the statement that:

$$\begin{aligned}\mathrm{I}[S_0; \overrightarrow{Y}_0] &= \mathrm{I}[\overleftarrow{Y}_0; \overrightarrow{Y}_0] \\ &\geq \mathrm{I}[R_0; \overrightarrow{Y}_0],\end{aligned}$$

where $R_0 \in \boldsymbol{\mathcal{R}}$. Moreover, among all equally predictive (*prescient rival*) partitions $\widehat{\boldsymbol{R}}$ of the past, the causal states minimize the state Shannon entropy: $\mathrm{H}[S] = \mathrm{H}[\pi] \leq \mathrm{H}[\widehat{R}]$. Due to the ϵ-machine's minimality, the *statistical complexity* $C_\mu = \mathrm{H}[S] = \mathrm{H}[\pi]$ measures the amount of historical information a process stores.

A process's ϵ-machine presentation has several additional properties that prove useful. First, the causal states form a Markov chain. This means that the ϵ-machine is a type of *hidden Markov model*. Second, the causal states are *unifilar*:

$$\mathrm{H}[S_{t+1} | Y_t, S_t] = 0.$$

That is, the current state and symbol uniquely determine the next state. This is necessary for an observer to maintain its knowledge of a process's current causal state while scanning a sequence of symbols. Third, unlike general (that is, nonunifilar) hidden Markov models, one can calculate a process's key informational properties directly from its ϵ-machine. For example, a process's *entropy rate* h_μ can be written in terms of the causal states:

$$h_\mu = \mathrm{H}[Y_0 | S_0].$$

And, using the methods of Refs. [14,20], a process's past-future mutual information—the *excess entropy* **E**—is given by its forward $\mathcal{S}^+$ and reverse $\mathcal{S}^-$ causal states:

$$\begin{aligned}\mathbf{E} &\equiv \mathrm{I}[\overleftarrow{Y}; \overrightarrow{Y}] \\ &= \mathrm{I}[\mathcal{S}^-; \mathcal{S}^+].\end{aligned} \tag{5}$$

Generally, the excess entropy is only a lower bound on the information that must be stored in order to predict a process: $\mathbf{E} \leq C_\mu$. This difference is captured by the process's *crypticity* $\chi = C_\mu - \mathbf{E}$.

Since they are conditioned on semi-infinite pasts, the causal states defined above correspond to *recurrent* states in a process's complete ϵ-machine presentation. They capture a process's time-asymptotic behavior. An ϵ-machine also has *transient* causal states that arise when conditioning on finite-length pasts, as well as a unique start state, which can be either transient or recurrent. When the underlying process is ergodic, they can be derived from the recurrent causal states using the mixed-state method of Ref. [14]. In general, they can be obtained from a modified causal equivalence relation, extended to include finite pasts. We omit them, unless otherwise noted, from our development.

The preceding introduced what is known as the *history* specification of an ϵ-machine: From a stationary, ergodic process, one derives its ϵ-machine by applying equivalence relation Eq. (4). There is also the complementary *generator* specification: As a generator, rather than using equivalence classes over a process's histories, the ϵ-machine is defined as a strongly connected hidden Markov model whose transitions are unifilar and whose states are probabilistically distinct. Such an ϵ-machine generates a unique, ergodic, stationary process, and is the same ϵ-machine that we would obtain by applying the causal equivalence relation to said generated process [21]. The following uses both history and generator ϵ-machines; which will be clear in context.

2.4 Example Process ϵ-Machines

The cardinality of a process's causal state set $\boldsymbol{\mathcal{S}}$ need not be finite or even countable; see, e.g., Figs. 7, 8, 10, and 17 in Ref. [22]. For simplicity in the following, though, we restrict our study to processes with a finite or countably infinite number of causal states. This allows us to depict a process graphically, showing its ϵ-machine as an *edge-labeled directed graph*. Nodes in the graph are causal states and edges, transitions between them. A transition from state σ_i to state σ_j while emitting symbol y is represented as an edge connecting the corresponding nodes that is labeled $\mathbf{y} : p$. (Anticipating our needs later on, this differs slightly from prior notation.) Here, $p = T_{ij}^{(y)}$ is the state transition probability and y is the symbol emitted on the transition. Figure 2 displays ϵ-machine state-transition diagrams for the example processes.

Since the Biased Coin Process's current output is independent of the past, all pasts are causally equivalent. This leads to a single causal state A, occupied with probability 1. (See Fig. 2a.) Therefore, the Biased Coin Process has a statistical complexity of $C_\mu = 0$ bits. It's excess entropy $\mathbf{E}$ also vanishes. This example illustrates the general property that IID processes lack causal structure. They can be quite random; the example here has an entropy rate of $h_\mu = \mathrm{H}(2/3) \approx 0.918$ bits per step, where $\mathrm{H}(p)$ is the binary entropy function.

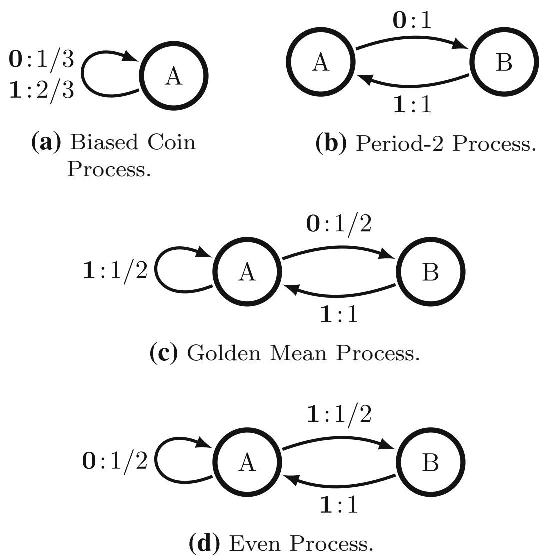


Fig. 2 ϵ-machines for the example processes. Transitions labeled $\mathbf{y} : p$, where $p = T_{ij}^{(y)}$ is the state transition probability and y is the symbol emitted on the transition

In contrast, the Period-2 Process has two causal states, call them A and B, that correspond to pasts that end in either a 1 or a 0, respectively. (See Fig. 2b.) Since the states are occupied with equal probability, the Period-2 Process has $C_\mu = 1$ bit of stored information. In this case, $\mathbf{E} = C_\mu$. It is perfectly predictable, with $h_\mu = 0$ bits per step.

The two causal states of the Golden Mean Process also correspond to pasts ending with either a 0 or 1, but for it the state transition structure ensures that no consecutive 0s are generated. (See Fig. 2c.) Causal state A is occupied with $\mathbb{P}(A) = 2/3$ and state B with $\mathbb{P}(B) = 1/3$, giving $C_\mu = \mathrm{H}(2/3) \approx 0.918$ bits. Note that the excess entropy is substantially less—$\mathbf{E} \approx 0.2516$ bits—indicating that the process is *cryptic*. We therefore must store additional state information above and beyond $\mathbf{E}$ in order to predict the process [20]. Its entropy rate is $h_\mu = \mathrm{H}(2/3) \approx 0.918$ bits per step.

As mentioned already, the Even Process cannot be represented by a finite Markov chain. It can be represented by a finite hidden Markov model, however. In particular, its ϵ-machine provides the most compact presentation—a two-state hidden Markov model. Causal state A corresponds to pasts that end with an even block of 1s, and state B corresponds to pasts that end with an odd block of 1s. (See Fig. 2d.) Since the probability distribution over states is the same as that of the Golden Mean Process, the Even process has $C_\mu \approx 0.918$ bits of statistical complexity and $h_\mu \approx 0.918$ bits per step. In contrast, the excess entropy and statistical complexity are equal: $\mathbf{E} = C_\mu$. The Even Process is not cryptic.

3 Input–Output Processes and Channels

Up to this point, we focused on a stochastic process and a particular canonical presentation of a mechanism—the ϵ-machine—that can generate it. We now turn to our main topic, generalizing the ϵ-machine presentation of a given process to a presentation of a controlled process—that is, to input–output (I/O) processes. I/O processes are related to probabilistic extensions of Moore's sequential machines [23,24], probabilistic extensions of codes from symbolic dynamics [25], and, perhaps more naturally, to Shannon's communication channels [26]. And so, we refer to them simply as *channels*. There are important differences from how channels are developed in standard treatments [12], though. The principal difference is that we consider channels with memory, while the latter in its elementary treatment, at least, typically considers memoryless channels or channels with restricted forms of memory [1]. In addition, with an eye to applications, the framework here is adapted to *reconstruct* channels from observations of an input–output process. (A topic to which we return at the end.) Finally, the development places a unique emphasis on detecting and analyzing structure inherent in I/O processes. Our development parallels that for ϵ-machines; see Ref. [1, and citations therein].

To begin, we define I/O processes. To make these concrete, we provide several example channels, leveraging the example processes and their ϵ-machines already described. The examples naturally lead to the desired extension—the ϵ-transducer. The development then turns to the main properties of ϵ-transducers.

Loosely speaking, a channel defines a coupling between stochastic processes. It can be memoryful, probabilistic, and even anticipate the future. In other words, the channel's current output symbol may depend probabilistically upon its past, current, and future input and output symbols. As such, we will be led to generalize the memoryless and anticipation-free communication channels primarily studied in elementary information theory [12].

[1] Though see Ref. [72, Ch.7] and for early efforts Refs. [73] and [74]

Definition 1 A *channel* $\overleftrightarrow{Y} \mid \overleftrightarrow{X}$ with *input alphabet* $\mathcal{X}$ and *output alphabet* $\mathcal{Y}$ is a collection of stochastic processes over alphabet $\mathcal{Y}$, where each such process $\overleftrightarrow{Y} \mid \overleftrightarrow{x}$ corresponds to a bi-infinite *input* sequence in $\overleftrightarrow{\mathcal{X}}$:

$$\overleftrightarrow{Y} \mid \overleftrightarrow{X} \equiv \{\overleftrightarrow{Y} \mid \overleftrightarrow{x}\}_{\overleftrightarrow{x} \in \overleftrightarrow{\mathcal{X}}} \tag{6}$$

That is, each fixed realization $\overleftrightarrow{x} = \ldots x_{-1}x_0x_1\ldots$ over input alphabet $\mathcal{X}$ is mapped to a stochastic process $\overleftrightarrow{Y} \mid \overleftrightarrow{x} = \overleftrightarrow{Y} \mid \ldots x_{-1}x_0x_1\ldots$ over alphabet $\mathcal{Y}$.

If we are given a process $\overleftrightarrow{X}$—an *input process*—then a channel maps this distribution over sequences to a joint process $\overleftrightarrow{(X, Y)}$, which can be marginalized to obtain a process $\overleftrightarrow{Y}$—the *output process*. We can characterize a channel by an indexed set $\mathcal{P}$ of *conditional word probabilities*:

$$\begin{aligned}
&\mathbb{P}(Y_{t:t+L} \mid \overleftrightarrow{x}) \\
&\quad \equiv \left\{\mathbb{P}\big(Y_{t:t+L} = y_{t:t+L} \big| \overleftrightarrow{X} = \overleftrightarrow{x}\big)\right\}_{y_{t:t+L} \in \mathcal{Y}^L,\, \overleftrightarrow{x} \in \overleftrightarrow{\mathcal{X}}} .
\end{aligned}$$

Equivalently, we can represent a channel as a distribution over bi-infinite sequences (output) conditioned on a particular bi-infinite sequence (input). In other words, a channel can be characterized by an indexed set:

$$\mathbb{P}\big(\overleftrightarrow{Y} \big| \overleftrightarrow{x}\big) \equiv \left\{\mathbb{P}\big(\overleftrightarrow{Y} \in \sigma \big| \overleftrightarrow{X} = \overleftrightarrow{x}\big)\right\}_{\sigma \subseteq \overleftrightarrow{\mathcal{Y}},\, \overleftrightarrow{x} \in \overleftrightarrow{\mathcal{X}}} .$$

Given an input process' distribution $\mathbb{P}(\overleftrightarrow{X})$ and a channel's distribution $\mathbb{P}\big(\overleftrightarrow{Y} \mid \overleftrightarrow{x}\big)$ for all inputs $\overleftrightarrow{x}$, we obtain the output process' distribution as follows:

$$\begin{aligned}
\mathbb{P}\big(\overleftrightarrow{Y}\big) &= \int \mathbb{P}\big(\overleftrightarrow{Y}, \overleftrightarrow{x}\big)\, \mathrm{d}\overleftrightarrow{x} \\
&= \int \mathbb{P}\big(\overleftrightarrow{Y} \mid \overleftrightarrow{x}\big)\mathbb{P}(\overleftrightarrow{x})\, \mathrm{d}\overleftrightarrow{x} ,
\end{aligned}$$

where the first integrand shows the appearance of the intermediate joint process $\overleftrightarrow{(X, Y)}$.

Let's say a few words about definitional choices made up to this point. As defined, the channels we consider are *total* (defined for every possible input sequence). One could extend the definition to allow for *partial* channels (defined only for a subset of possible sequences), but we do not consider such channels in what follows. This is the primary reason for defining a channel's domain in terms of bi-infinite sequences rather than say, collections of finite words. Such channels would not necessarily be total. Also, requiring that channels be defined for *every finite* input word is restrictive, as even the simplest channels may not have well defined behavior for say, a single symbol input word. We could instead define channels over a *subset* of all finite input words and explicitly add in the restriction that the channel be total, but this is still more restrictive than the $\overleftrightarrow{\mathcal{X}}$ definition above. Consider, for example, the channel that outputs all 1s if its input sequence contains at least one 1 and outputs all 0s otherwise. Such a channel is undefined for any finite input word that consists of all 0s, but is well defined for any bi-infinite binary sequence. It is also true that any total channel defined over finite input words can be trivially defined over bi-infinite sequences by appending an arbitrary infinite past and future to each finite word (without changing the channel's output behavior).

Stationarity is as useful a property for channels as it is for processes.

Definition 2 A channel is *stationary* if and only if its probability distributions are invariant under time translation:

$$\mathbb{P}(Y_{t:t+L} | \overleftrightarrow{x_t}) = \mathbb{P}(Y_{0:L} | \overleftrightarrow{x_0}) \text{ and}$$
$$\mathbb{P}(\overleftrightarrow{Y_t} | \overleftrightarrow{x_t}) = \mathbb{P}(\overleftrightarrow{Y_0} | \overleftrightarrow{x_0}),$$

for all $t \in \mathbb{Z}$, $L \in \mathbb{Z}^+$, and every input sequence $\overleftrightarrow{x}$.

An immediate consequence is that stationary channels map stationary input processes to stationary output processes:

Proposition 1 (Stationarity Preservation) *When a stationary channel is applied to a stationary input process, the resulting output process is also stationary.*

Proof One calculates directly:

$$\begin{aligned}
\mathbb{P}(\overleftrightarrow{Y_t}) &= \int \mathbb{P}(\overleftrightarrow{Y_t} | \overleftrightarrow{x_t}) \mathbb{P}(\overleftrightarrow{x_t}) \mathrm{d}\overleftrightarrow{x_t} \\
&= \int \mathbb{P}(\overleftrightarrow{Y_0} | \overleftrightarrow{x_0}) \mathbb{P}(\overleftrightarrow{x_0}) \mathrm{d}\overleftrightarrow{x_0} \\
&= \mathbb{P}(\overleftrightarrow{Y_0}),
\end{aligned}$$

where the second equality follows from stationarity of both input process and channel, once we note that $\ldots \mathrm{d}x_{t-1} \mathrm{d}x_t \mathrm{d}x_{t+1} \ldots = \ldots \mathrm{d}x_{-1} \mathrm{d}x_0 \mathrm{d}x_1 \ldots$, since under shifted indexing the relationships between x_i and infinitesimals $\mathrm{d}x_i$ are preserved and the latter themselves do not change. □

We also primarily restrict our discussion to ergodic channels [27].

Definition 3 An *ergodic channel* is a channel that maps any ergodic input process $\overleftrightarrow{X}$ to an ergodic joint process $\overleftrightarrow{(X, Y)}$.

Another important channel property is that of causality.

Definition 4 A *causal channel* is anticipation-free:

$$\mathbb{P}(Y_{t:t+L} | \overleftrightarrow{X}) = \mathbb{P}(Y_{t:t+L} | \overleftarrow{X}_{t+L}).$$

That is, the channel has well defined behavior on semi-infinite input pasts and is completely characterized by that behavior.

Channel causality is a reasonable assumption when a system has no access to future inputs. However, as a note of caution in applying the following to analyze, say, large-scale systems with many components, the choice of observables may lead to input–output processes that violate causality. For example, treating spatial configurations of one-dimensional spin systems or cellular automata as if they were time series—a somewhat common strategy—violates causality. In the following, though, we assume channel stationarity and causality, unless stated otherwise.

It is worth noting that causality is often not a severe restriction. Specifically, the following results extend to channels with finite anticipation—channels whose current output depends upon N future inputs. When both the input process and channel are stationary, one delays the appearance of the channel output by N time indices. This does not change the output process, but converts *finite anticipation* to finite channel memory and delayed output. In this way, it is possible to apply the analysis to follow to channels with anticipation directly, but the optimality theorems established must be modified slightly.

4 Example Channels and Their Classification

To motivate our main results, consider several example stationary causal channels with binary input and output. Figure 3 illustrates the mapping from input words $\{x_{0:L} = x_0 x_1 \ldots x_{L-1}\}$ to output words $\{y_{0:L} = y_0 y_1 \ldots y_{L-1}\}$ for the example channels. The *wordmaps* there treat each input–output pair $(x_{0:L}, y_{0:L})$ as a point $p = (p_x, p_y)$ in the unit square, where the input (output) word forms the binary expansion of $p_x = 0.x_0 x_1 \cdots x_{L-1}$ ($p_y = 0.y_0 y_1 \cdots y_{L-1}$). Since the channels are defined for all inputs, the plots show every input word.

We organize the examples around a classification scheme paralleling that used in signal processing to highlight memory and the nature of feedback [28].

We describe channel behavior using recurrence relations of the form:

$$Y_t \sim r\,(\overleftarrow{X}_t, \overleftarrow{Y}_t, X_t)\,,$$

where $r(\cdot)$ is either a logical function, in simple cases, or a distribution. Note that the next output Y_t depends only on the past—the channels are causal.

In general, such a recurrence relation only captures the channel's *present* behavior. Fortunately, a causal channel's future behavior is summarized by its present behavior:

$$\begin{aligned}
&\mathbb{P}(Y_{t:t+L}|\overleftarrow{(X,Y)}_t, X_{t:t+L}) \\
&\quad = \mathbb{P}(Y_t|\overleftarrow{(X,Y)}_t, X_{t:t+L})\mathbb{P}(Y_{t+1}|\overleftarrow{(X,Y)}_t, X_{t:t+L}, Y_t)\cdots\mathbb{P}(Y_{t+L-1}|\overleftarrow{(X,Y)}_t, X_{t:t+L}, Y_{t:t+L-1}) \\
&\quad \overset{(a)}{=} \mathbb{P}(Y_t|\overleftarrow{(X,Y)}_t, X_{t:t+L})\mathbb{P}(Y_{t+1}|\overleftarrow{(X,Y)}_{t+1}, X_{t+1:t+L})\cdots\mathbb{P}(Y_{t+L-1}|\overleftarrow{(X,Y)}_{t+L-1}, X_{t+L-1}) \\
&\quad \overset{(b)}{=} \mathbb{P}(Y_t|\overleftarrow{(X,Y)}_t, X_t)\mathbb{P}(Y_{t+1}|\overleftarrow{(X,Y)}_{t+1}, X_{t+1})\cdots\mathbb{P}(Y_{t+L-1}|\overleftarrow{(X,Y)}_{t+L-1}, X_{t+L-1}) \\
&\quad \overset{(c)}{=} \mathbb{P}(Y_t|\overleftarrow{(X,Y)}_t, X_t)\mathbb{P}(Y_t|\overleftarrow{(X,Y)}_t, X_t)\cdots\mathbb{P}(Y_t|\overleftarrow{(X,Y)}_t, X_t) \\
&\quad = \mathbb{P}(Y_t|\overleftarrow{(X,Y)}_t, X_t)^L,
\end{aligned}$$

where in (a) we merge individual variables into pasts to obtain new pasts, in (b) we remove input variables that have no effect due to causality, and (c) follows from stationarity.

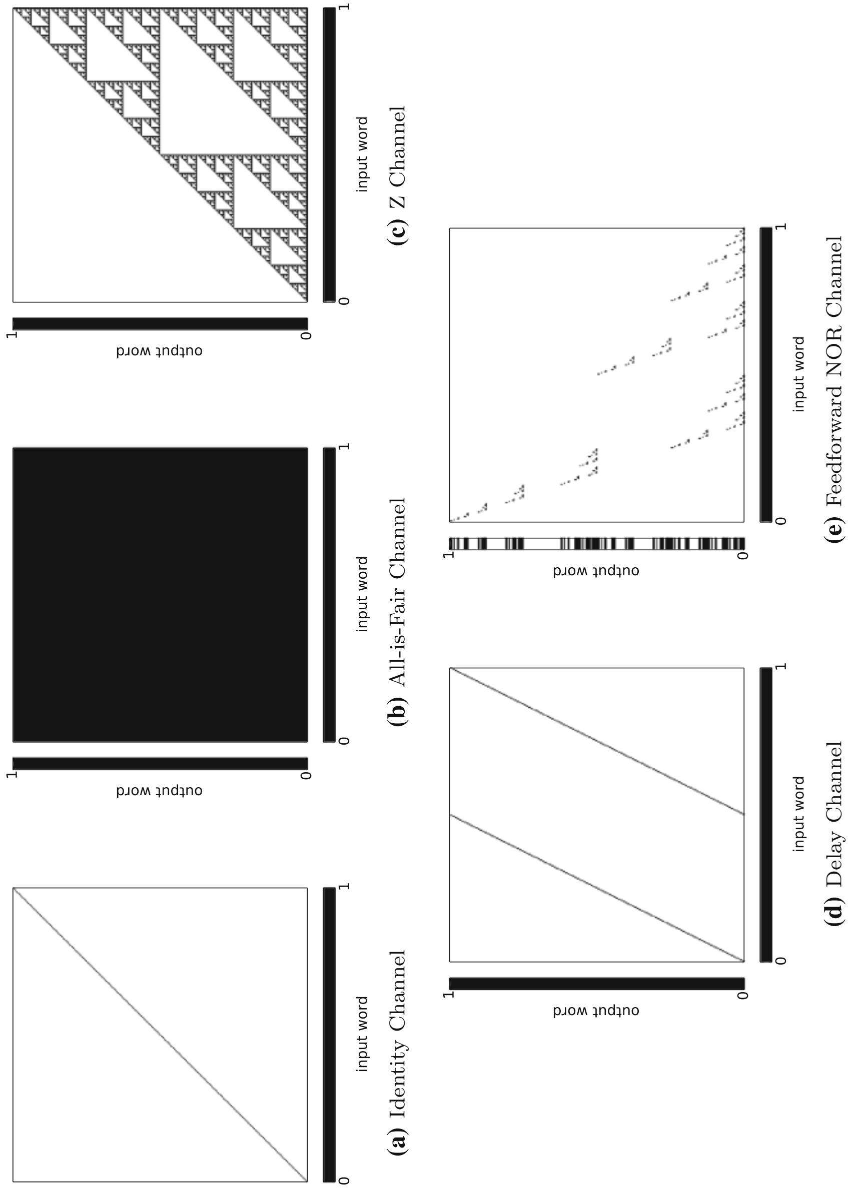
4.1 Memorylessness

The *Memoryless Binary Channel*'s (MBC's) current output depends only on its current input; the analog of an IID process in that its behavior at time t is independent of that at other times. The MBC includes as special cases the *Binary Symmetric Channel* (BSC) and the *Z Channel* [12]. We can summarize the MBC's behavior with a simplified recurrence relation of the form $Y_t \sim r\,(X_t)$ and its conditional probabilities factor as follows:

$$\mathbb{P}(Y_{t:t+L}|\overleftrightarrow{X}) = \mathbb{P}(Y_t|X_t)\mathbb{P}(Y_{t+1}|X_{t+1})\cdots\mathbb{P}(Y_{t+L-1}|X_{t+L-1}).$$

The first three wordmaps of Fig. 3 illustrate the behavior of memoryless channels. We see that the *Identity Channel* (Fig. 3a) always maps a word to itself. Whereas, the *All-is-Fair Channel* (Fig. 3b) maps each input word uniformly to every output word. We immediately see that deterministic channels have wordmaps with a single filled pixel per plot column.

The Z Channel wordmap is shown in Fig. 3c. It transmits all 0s with no noise, but adds noise to all 1s transmitted. The wordmap shows the maximal noise case, where all 1s are replaced with the output of a fair coin. We see that even memoryless channels have nontrivial word mappings. In this case, the latter forms a self-similar Sierpinski right triangle [29] in the unit square.



(a) Identity Channel

(b) All-is-Fair Channel

(c) Z Channel

(d) Delay Channel

(e) Feedforward NOR Channel

Fig. 3 Wordmaps for simple example channels display the map from finite input words to finite output words: Allowed input–output pairs $(x_{0:8}, y_{0:8})$ correspond to a dot placed in the unit square at $(0.x_0x_1\cdots x_7, 0.y_0y_1\cdots y_7)$. The possible observed input (output) words are displayed below (left) of the horizontal (vertical) axis

4.2 Finite Feedforward

Now, consider channels whose behavior depends only on a finite input history. Their behavior on input is analogous to those of order-R Markov chains. They can also be thought of as stochastic, anticipation-free *sliding block codes* [25] with finite memory or as generalized *finite impulse response filters* [28]. These channels' behavior can be summarized with a recurrence relation of the form $Y_t \sim r(X_{t-M:t}, X_t)$, where M is a finite input history length such that $M > 0$.

The *Delay Channel* simply stores its input at time $t-1$ and outputs it at time t. Its wordmap is shown in Fig. 3d. Those familiar with one-dimensional iterated maps of the interval will recognize the word mapping as the *shift map*, capturing the fact that delayed output corresponds to a binary shift applied to the input word. Note that when viewed as a function on the space of processes, the Delay Channel acts as the identity.

The *Feedforward NOR Channel*'s output at time t results from performing a logical NOR ($\downarrow$) on its inputs at times t and $t-1$:

$$y_t = x_{t-1} \downarrow x_t .$$

The wordmap for the Feedforward NOR Channel is shown in Fig. 3e. There, we see that although the channel is deterministic, the wordmap's self-similarity makes it difficult to see that the mapping is a function—that there is, in fact, a single output word for each input.

The additional complexity of the remaining examples does not lead to new types of apparent graphical structure beyond that seen in the existing wordmaps. So, wordmaps will be omitted for now. With additional theory developed, we return to illustrate these channels, but using a more advanced form of wordmap.

4.3 Infinite Feedforward

Generally, channels depend on infinitely long input histories. This behavior is analogous to the long-range dependence seen in typical hidden Markov models [30,31] or in *strictly sofic* subshifts [25]. Channels with dependence upon infinitely long input histories *alone* can also be interpreted as generalized *infinite impulse response filters* [28]. The behavior of such channels can be summarized by a recurrence relation of the form $Y_t \sim r(\overleftarrow{X_t}, X_t)$.

The *Odd NOT* Channel stores the parity (even or odd) of the number of ones observed in its input since the last zero observed; much like the Even Process. If the parity is currently even, it behaves as the Identity Channel. If the parity is odd, it outputs the bitwise NOT (bit flip) of its input. Since the channel's behavior depends on the parity of its input, it cannot be characterized by finite input histories alone.

A channel can depend, however, on past *outputs* as well as inputs. Such *feedback* can allow one to replace the infinite-history recurrence relation with one that includes only a finite history of inputs and outputs. Consider again the Odd NOT Channel described above. Note that its behavior is determined entirely by the current input value, as well as the parity of the number of ones observed on input. In fact, the parity at time t can be summarized by the input and output at time $t-1$. If $x_{t-1} = 0$, the parity will always be even. If $x_{t-1} = 1$, and $y_{t-1} = 0$ we know that the parity *was* odd, since the bit was flipped, but since a 1 was just observed on input, the parity is now even. Finally, if $x_{t-1} = 1$ and $y_{t-1} = 1$, we know that the parity was previously even, and the newly observed 1 makes the current parity *odd*. Summarizing, we have that:

$$(x, y)_{t-1} = (0, 0) \Leftrightarrow \text{Even input history parity,}$$
$$(x, y)_{t-1} = (0, 1) \Leftrightarrow \text{Even input history parity,}$$

$$(x, y)_{t-1} = (1, 0) \Leftrightarrow \text{Even input history parity, and}$$
$$(x, y)_{t-1} = (1, 1) \Leftrightarrow \text{Odd input history parity.} \quad (7)$$

By allowing feedback, we can therefore summarize the behavior of the Odd NOT Channel with a recurrence relation that depends only upon finite history (of length $M = 1$): $Y_t \sim r(X_{t-1}, Y_{t-1}, X_t)$.

An interesting observation is that the Odd NOT channel maps the Even Process to a bit-flipped Golden Mean Process. However, a single process-to-process mapping does *not* uniquely define a channel. There are an infinite number of channels, in fact, that map the Even Process to the bit-flipped Golden Mean Process.

4.4 Finite Feedback

As seen in the previous section, allowing for even a finite amount of output feedback can can lead to substantial simplifications in the description of a channel. Let's consider channels that depend on a finite output history on their own terms.

The most trivial case would be channels that depend solely on output histories, with no dependence on inputs. Since there is effectively no input, these channels reduce to the output-only stochastic processes (generators) discussed earlier. Consider, for example, the *All is Golden Channel* that outputs the Golden Mean Process, regardless of what input it receives.

A less trivial example is the *Feedback NOR Channel*, similar to the Feedforward NOR Channel, except the output at time t is the logical NOR of its current input and *previous output*:

$$y_t = x_t \downarrow y_{t-1}.$$

This channel's behavior is clear in this feedback form. It might be desirable, however, to find a purely feedforward presentation for the channel. The recurrence relation for the Feedback NOR Channel can be solved recursively to give a feedforward presentation that is defined for almost every input history. We recurse in the following way:

$$\begin{aligned} y_t &= x_t \downarrow y_{t-1} \\ &= x_t \downarrow (x_{t-1} \downarrow y_{t-2}) \\ &= x_t \downarrow (x_{t-1} \downarrow (x_{t-2} \downarrow y_{t-3})) \\ &= \cdots , \end{aligned}$$

and so on, until reaching sufficiently far into the input past that a 1 is observed. When this happens, the recursion terminates as the output of the NOR function is always 0 when either argument is 1. We can therefore construct a purely feedforward recurrence relation, but the input histories can be arbitrarily long—corresponding to arbitrarily long input histories consisting entirely of 0s. The resulting feedforward recurrence relation is defined for all histories, except for the infinite history of all 0s.

Note that such ill-defined behavior for certain infinite histories is typical in systems that have infinite memory lengths and is not a problem specific to channels. One can be careful to explicitly *define* behavior for such cases, but this is beyond the scope of our current work, and these pathological histories typically occur with zero probability.

In contrast, consider replacing the logical NOR in the Feedback NOR Channel with an exclusive OR (XOR or $\oplus$), thus giving the *Feedback XOR Channel*:

$$y_t = x_t \oplus y_{t-1}. \tag{8}$$

In this case, solving for a pure-feedforward relation fails since the output of a logical XOR is never determined by a single argument. This channel illustrates the fact that a presentation which includes feedback cannot always be reduced to a pure-feedforward presentation.

4.5 Infinite Feedback

Just as we can define channels whose behavior depends upon infinite input histories, we can define channels whose behavior depends upon infinite *output* histories. In fact, we have already studied a channel that can be represented this way. Consider a channel that stores the parity (even or odd) of the number of ones observed in its *output* since the last zero observed. If this parity is currently even, it behaves as the Identity Channel. If the parity is odd, it outputs the bitwise NOT (bit flip) of its input. This appears to be very similar to the Odd NOT Channel defined above, but with a dependence on infinite output histories and the present input, rather than infinite input histories. In fact, this is simply a different *presentation* of the Odd NOT Channel.

It suffices to show that the feedback presentation of the Odd NOT Channel can be reduced to the same finite history presentation as with its feedforward presentation. Proceeding as before, we observe that if $y_{t-1} = 0$, the output parity is always even. If $x_{t-1} = 0$ and $y_{t-1} = 1$, the previous parity was odd (the bit was flipped), but the 1 observed on output makes the output parity even. Finally, if $x_{t-1} = 1$ and $y_{t-1} = 1$, the output parity was even, and the 1 observed makes the output parity odd. Summarizing, we obtained the same presentation as the feedforward presentation specified by Eq. (7).

4.6 Infinite Feedforward-Feedback

We just examined an example channel whose presentation depends upon infinite histories when only feedforward or feedback is allowed, but only a finite history when both feedforward and feedback are allowed. The following example shows that finding a finite history presentation is not always possible. Channels of this form are perhaps the most natural channel generalization of infinite Markov order (strictly sofic) processes.

The *Odd Random Channel* stores the parity of its input history just as the Odd NOT Channel does, and it again behaves as the identity when the parity is currently even. When the parity is odd, the channel outputs a 0 or 1 with equal probability. Like the Odd NOT Channel, this has an infinite feedback presentation that stores the channel's output history parity. The channel does *not* have any finite history presentation, however. If one attempts to construct a finite presentation via the recursion unrolling procedure used for the Odd NOT Channel, it is simple to obtain the following relationships:

$$\begin{aligned}
(x, y)_{t-1} &= (0, 0) \Leftrightarrow \text{Even input history parity,} \\
(x, y)_{t-1} &= (0, 1) \Leftrightarrow \text{Even input history parity, and} \\
(x, y)_{t-1} &= (1, 0) \Leftrightarrow \text{Even input history parity.}
\end{aligned}$$

The problem arises from the fact that when $x_{t-1} = 1$ and $y_{t-1} = 1$, the input history parity is uncertain. The channel could have been operating as the identity (even parity) or giving random output (odd parity). Looking at progressively longer histories can resolve this uncertainty, but only once a 0 has been observed (on either input, output, or both). This ambiguity requires that we specify arbitrarily long joint histories to determine the behavior of the channel in general. It is therefore not possible to construct any finite-history presentation of the Odd Random Channel.

4.7 Irreducible Feedforward-Feedback

As a final example, consider a channel that has a finite presentation when both feedforward and feedback are allowed, but has *no* pure-feedforward or pure-feedback presentation. The *Period-2 Identity NOT Channel* alternates between the identity and bit flipped identity at each time step. This channel's present behavior is completely determined by whether the previous input and output bits, x_{t-1} and y_{t-1}, match. When the bits match, the channel was in its "identity" state and is therefore now in its "NOT" state. The opposite is clearly true when the bits do not match. The channel therefore has a recurrence relation of the form $Y_t \sim r(X_{t-1}, Y_{t-1}, X_t)$.

Since the behavior does not depend on the particular values of the input or output, but whether or not they *match*, there is no way to construct a pure-feedforward or pure-feedback presentation of the channel. This channel therefore illustrates the notion of *irreducible* output (or input) memory—dependence upon past output (input) that cannot be eliminated even by including dependence upon infinite past outputs (inputs).

4.8 Causal Channel Markov Order Hierarchy

It turns out that the set of examples above outlines a classification scheme for causal channels in terms of their Markov orders [32] that we now make explicit. Just as Markov order plays a key role in understanding the organization of processes, it is similarly helpful for channels. Channel Markov orders are the history lengths required to *completely* specify a causal channel's behavior, given certain constraints on knowledge of other histories.

Definition 5

1. The *pure feedforward Markov order R_{pff}* is the smallest M such that $Y_t \sim r\,(X_{t-M:t}, X_t)$; i.e., $\mathbb{P}\,(Y_t|\overleftarrow{(X,Y)}_t, X_t) = \mathbb{P}\,(Y_t|X_{t-M:t}, X_t)$.
2. The *pure feedback Markov order R_{pfb}* is the smallest M such that $Y_t \sim r\,(Y_{t-M:t}, X_t)$; i.e., $\mathbb{P}\,(Y_t|\overleftarrow{(X,Y)}_t, X_t) = \mathbb{P}\,(Y_t|Y_{t-M:t}, X_t)$.
3. The *channel Markov order R* is the smallest M such that $Y_t \sim r\,(X_{t-M:t}, Y_{t-M:t}, X_t)$; i.e., $\mathbb{P}\,(Y_t|\overleftarrow{(X,Y)}_t, X_t) = \mathbb{P}\,(Y_t|X_{t-M:t}, Y_{t-M:t} X_t)$.
4. The *irreducible feedforward Markov order R_{iff}* is the smallest M such that $Y_t \sim r\,(X_{t-M:t}, \overleftarrow{Y}_t, X_t,)$; i.e., $\mathbb{P}\,(Y_t|\overleftarrow{(X,Y)}_t, X_t) = \mathbb{P}\,(Y_t|X_{t-M:t}, \overleftarrow{Y}_t, X_t)$.
5. The *irreducible feedback Markov order R_{ifb}* is the smallest M such that $Y_t \sim r\,(\overleftarrow{X}_t, Y_{t-M:t}, X_t)$; i.e., $\mathbb{P}\,(Y_t|\overleftarrow{(X,Y)}_t, X_t) = \mathbb{P}\,(Y_t|\overleftarrow{X}_t, Y_{t-M:t}, X_t)$.

For example, we showed that the Odd NOT Channel's presentation requires an infinite history when only feedforward or feedback is allowed. And so, it has $R_{\mathrm{pff}} = R_{\mathrm{pfb}} = \infty$. However, it only requires finite history when both are allowed: $R = 1$. If we have full knowledge of the output past, we still need one symbol of input history in order to characterize the channel, so $R_{\text{iff}} = 1$. Similarly, we have $R_{\text{ifb}} = 1$.

Note that the irreducible feedforward Markov order R_{iff} will only be nonzero if the pure feedback order R_{pfb} is undefined. Similarly, the irreducible feedback Markov order R_{ifb} will only be nonzero when the pure feedforward order R_{pff} is undefined. In words, if a channel has irreducible feedback (feedforward), the channel has no pure feedforward (feedback) presentation. Moreover, the channel Markov order R bounds the pure Markov orders from below and the smallest of the two irreducible Markov orders bounds the channel Markov order from below:

$$\min(R_{\text{iff}}, R_{\text{ifb}}) \leq R \leq \min(R_{\text{pff}}, R_{\text{pfb}}).$$

In a sequel, we address input and output memory using information-theoretic quantities. There, we characterize different *amounts* of input and output memory and how they relate, whereas here in discussing Markov orders we considered *lengths* of input and output sequences. We also focused on causal channels, which allowed us to restrict our discussion to present behavior and memory lengths. In the case of anticipatory channels, we must explicitly consider *future* behavior, as well as *anticipation* lengths. However, this is best left to another venue, so that we do not deviate too far from the path to our goal.

4.9 Causal-State Channels

There is a natural and quite useful channel embedded in any ϵ-machine presentation of a stationary process—*the causal-state channel*—that identifies structure embedded in a process via the ϵ-machine's causal states.

Consider a process and its ϵ-machine M. Previously, we described M as a generator of the process. An ϵ-machine, however, is also a *recognizer* of its process's sequences. Briefly, M reads a sequence and follows the series of transitions determined by the symbols it encounters. The output of the causal-state channel is then the sequence of causal states. The operation of this channel is what we call *causal-state filtering*. Notably, the induced mapping is a function due to the ϵ-machine's unifilarity.

In this way, the channel filters observed sequences, returning step-by-step associated causal states. Given an ϵ-machine, the causal-state filter has the same topology as the ϵ-machine, but input symbols match the ϵ-machine transition symbols and output symbols are the state to which the transition goes.

The recursion relation for causal-state filtering is:

$$S_t \sim r\left(X_{t-1}, S_{t-1}\right).$$

As just noted, $r(\cdot)$ is a (nonprobabilistic) function, determined by the ϵ-map:

$$\begin{aligned} S_t &= \epsilon(\overleftarrow{X}_t) \\ &= \epsilon(X_{t-1}, \overleftarrow{X}_{t-1}) \\ &= \epsilon(X_{t-1}, S_{t-1}). \end{aligned}$$

Thus, the pure feedback order is $R_{\text{pfb}} = 1$, as is the channel Markov order $R = 1$. The pure feedforward order R_{pff}, however, is the original process's Markov order. For methods to determine the latter see Ref. [32].

In this way, an ϵ-machine can be used to detect the hidden structures captured by the causal states. For example, causal-state filtering has been used to great effect in detecting emergent domains, particles, and particle interactions in spatially extended dynamical systems [33], single-molecule conformational states [34], and polycrystalline and fault structures in complex materials [35].

5 Global ϵ-Machine

Before introducing channel presentations, we first frame the question in the global setting of the ϵ-machine of joint (input–output) processes. This, then, grounds the ϵ-transducer in terms of an ϵ-machine.

Given a stationary process $\overleftrightarrow{X}$ and a channel $\overleftrightarrow{Y} \big| \overleftrightarrow{X}$ with output process $\overleftrightarrow{Y}$, form a joint random variable $Z_t = (X, Y)_t$ over input–output symbol pairs with alphabet $\boldsymbol{\mathcal{Z}} = \boldsymbol{\mathcal{X}} \times \boldsymbol{\mathcal{Y}}$. For example, if $\boldsymbol{\mathcal{X}} = \{a, b\}$ and $\boldsymbol{\mathcal{Y}} = \{c, d\}$, then $\boldsymbol{\mathcal{X}} \times \boldsymbol{\mathcal{Y}} = \{ac, ad, bc, bd\}$.

Definition 6 The process $\overleftrightarrow{Z}$ over $\boldsymbol{\mathcal{Z}}$ defines the channel's *I/O process*.

Definition 7 A stationary channel's *global* ϵ-machine is the ϵ-machine of its I/O process.

In this setting, the next section asks for a particular decomposition of the global ϵ-machine, when one has selected a portion of Z_t as "input" and another as "output". The input process $\overleftrightarrow{X}$ is then described by the marginal distribution of the joint process that projects onto "input" sequences. The same also holds for the output process $\overleftrightarrow{Y}$. In this way, the following results not only provide an analysis for specified input and output processes, but also an analysis of possible input-to-output mappings embedded in any process or its ϵ-machine. Leveraging this observation and anticipating the sequels, we also note here that the global ϵ-machine also provides the proper setting for posing questions about information storage and flow *within* any given process.

6 ϵ-Transducer

Computational mechanics' fundamental assumption—only prediction matters—applies as well to channels as to processes. In the case of channels, though, we wish to predict the channel's future *output* given the channel's past inputs *and* outputs *and* the channel's future *input*. This leads to a new causal equivalence relation $\sim_\epsilon$ over *joint* pasts $\overleftarrow{z} = \overleftarrow{(x, y)}$:

$$
\begin{aligned}
\overleftarrow{(x, y)} \sim_\epsilon \overleftarrow{(x, y)}' \iff & \\
& \mathbb{P}\big(\overrightarrow{Y} \big| \overrightarrow{X}, \overleftarrow{(X, Y)} = \overleftarrow{(x, y)}\big) \\
& \quad = \mathbb{P}\big(\overrightarrow{Y} \big| \overrightarrow{X}, \overleftarrow{(X, Y)} = \overleftarrow{(x, y)}'\big).
\end{aligned}
\tag{9}
$$

Compare Eq. (4) applied to the I/O process. The equivalence classes of $\sim_\epsilon$ partition the set $\overleftarrow{\boldsymbol{\mathcal{Z}}} = \overleftarrow{(\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{Y}})}$ of all input–output pasts. These classes are the channel's *causal states*, denoted $\boldsymbol{\mathcal{S}}$. The ϵ-*map* is a function $\epsilon : \overleftarrow{(\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{Y}})} \to \boldsymbol{\mathcal{S}}$ that maps each joint past to its corresponding channel causal state or, equivalently, to the set of joint pasts to which it is causally equivalent:

$$
\epsilon\big(\overleftarrow{(x, y)}\big) = \sigma_i = \big\{\overleftarrow{(x, y)}' : \overleftarrow{(x, y)} \sim_\epsilon \overleftarrow{(x, y)}'\big\}.
$$

The dynamic over causal states is again inherited from the implicit dynamic over joint pasts via the ϵ-map, resulting from appending the joint symbol $z_t = (x, y)_t$, as shown in

$$\begin{tikzcd}
\cdots \arrow[r, "(x,y)_{t-2}"] & \overleftarrow{(x,y)}_{t-1} \arrow[r, "(x,y)_{t-1}"] \arrow[d, "\epsilon"] & \overleftarrow{(x,y)}_{t} \arrow[r, "(x,y)_{t}"] \arrow[d, "\epsilon"] & \overleftarrow{(x,y)}_{t+1} \arrow[r, "(x,y)_{t+1}"] \arrow[d, "\epsilon"] & \cdots \\
\cdots \arrow[r, "(x,y)_{t-2}"] & s_{t-1} \arrow[r, "(x,y)_{t-1}"] & s_{t} \arrow[r, "(x,y)_{t}"] & s_{t+1} \arrow[r, "(x,y)_{t+1}"] & \cdots
\end{tikzcd}$$

Fig. 4 ϵ-Transducer dynamic induced by the causal states $s_t = \sigma_i \in \boldsymbol{\mathcal{S}}$ from a channel's joint pasts via the ϵ-map: $s_t = \epsilon(\ldots(x, y)_{t-2}(x, y)_{t-1}) \xrightarrow{(x,y)_t} s_{t+1} = \epsilon(\ldots(x, y)_{t-2}(x, y)_{t-1}(x, y)_t)$

Fig. 4. Since state transitions now depend upon the current input symbol, we specify the dynamic by an indexed set of *conditional-symbol transition matrices*:

$$\boldsymbol{\mathcal{T}} \equiv \left\{T^{(y|x)}\right\}_{x\in\boldsymbol{\mathcal{X}},y\in\boldsymbol{\mathcal{Y}}},$$

where $T^{(y|x)}$ has elements:

$$T_{ij}^{(y|x)} = \mathbb{P}(S_1 = \sigma_j, Y_0 = y|S_0 = \sigma_i, X_0 = x).$$

While the causal states for a stationary process have a *unique* stationary distribution, each stationary input to a channel can drive its causal states into a *different* stationary state distribution. The ϵ-map from joint histories to the channel's causal states can also be seen as a function that maps a *distribution* over joint histories to a distribution over the channel's causal states. Since each input history specifies a particular distribution over *output* histories (via the channel's conditional word probabilities), it follows that a *distribution* over input histories also specifies a distribution over output histories. When this input history distribution is specified via a particular input process, we obtain a unique distribution over causal states via its $\epsilon(\cdot)$ function. We write this input-dependent state distribution π_X as:

$$\begin{aligned}\pi_X(i) &= \mathbb{P}_X(S_0 = \sigma_i)\\ &= \mathbb{P}_X\big(\epsilon\big(\overleftarrow{(X,Y)}\big) = \sigma_i\big),\end{aligned}$$

where the subscript X indicates that the input process has a specific, known distribution. The distribution over joint histories is stationary by assumption here and, since the ϵ-map is time independent, π_X is stationary. We therefore refer to π_X as the (input-dependent) *stationary distribution*.

When both the input process and channel are stationary and ergodic, we can calculate this stationary distribution from the input and channel's causal-state transition matrices using a generalization of the algorithm found in Ref. [17]. We save an in-depth discussion of this algorithm for a sequel.

Definition 8 The tuple $(\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{Y}}, \boldsymbol{\mathcal{S}}, \boldsymbol{\mathcal{T}})$—consisting of the channel's input and output alphabets, causal states, and conditional-symbol transition probabilities, respectively—is the channel's ϵ-transducer.

Note that in the causal equivalence relation for channels, we condition on the input future $\overrightarrow{X}$, as a channel is defined by its output behavior *given* input. Requiring causal equivalence for the output future alone (or the joint future for that matter) requires knowledge of a particular *input process* as well. In particular, if we *do* have knowledge of the input process we can extend the standard causal equivalence relation to an equivalence relation involving

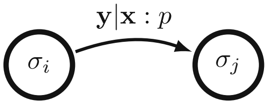


Fig. 5 Generic transition from ϵ-transducer causal state σ_i to state σ_j while accepting input symbol x and emitting symbol y with probability $p = T_{ij}^{(y|x)}$

joint pasts and joint future morphs, giving us the *global* (joint) ϵ-machine of the preceding section.

As with ϵ-machines, it is useful to consider channels whose ϵ-transducers have a finite (or countable) number of causal states. This restriction again allows us to represent an ϵ-transducer as a labeled-directed graph. Since transitions between causal states now depend on inputs as well as outputs, we represent a transition from state σ_i to state σ_j while accepting input symbol x and emitting symbol y as a directed edge from node σ_i to node σ_j with edge label $\mathbf{y}|\mathbf{x} : p$, where $p = T_{ij}^{(y|x)}$ is the edge transition probability. This is illustrated in Fig. 5.

7 Structural Complexity

To monitor the degree of structuredness in an ϵ-transducer, we use the Shannon information captured by its causal states, paralleling the definition of ϵ-machine statistical complexity. The stationary distribution over ϵ-transducer states allows one to define an ϵ-transducer's *input-dependent* statistical complexity:

$$C_X = \mathrm{H}[\pi_X].$$

(Note that X replaces the previously subscripted measure μ in ϵ-machine statistical complexity to specify the now-relevant measure.) While quantifying structural complexity, C_X's dependence on input requires a new interpretation. Some processes drive a transducer into simple, compressible behavior, while others will lead to complex behavior. Figure 6 illustrates this.

Input dependence can be removed by following the standard definition of channel capacity [12], giving a single number characterizing an ϵ-transducer. We take the supremum of the statistical complexity over input processes. This gives an upper bound on ϵ-transducer complexity—the *channel complexity*:

$$\overline{C_\mu} = \sup_X C_X,$$

where the maximizing input measure μ is implicitly defined. Note that not all transducers can be driven to a uniform distribution over states. Thus, recalling that uniform distributions maximize Shannon entropy, in general $\overline{C_\mu} \leq C_0 \equiv \log_2 |\boldsymbol{\mathcal{S}}|$—the *topological state complexity*.

8 Reproducing a Channel

To establish that the ϵ-transducer is an exact presentation of the causal channel it models, we must show that it reproduces the channel's conditional word probabilities. We first establish some needed notation.

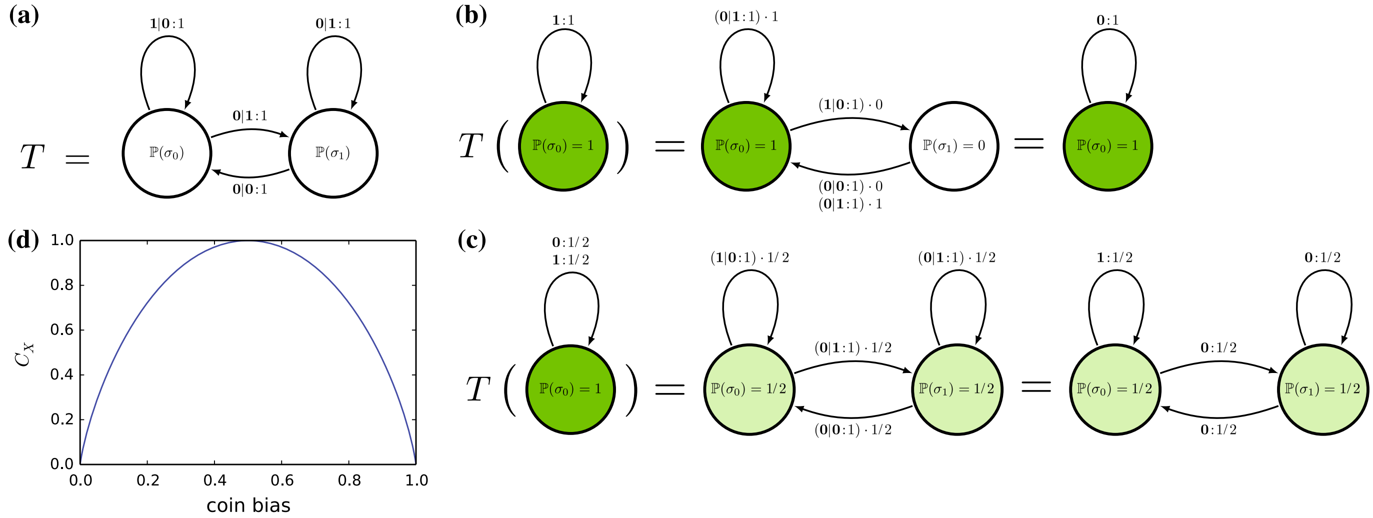


Fig. 6 A channel's statistical complexity C_X depends on the input process X: **a** an example transducer T—the feedforward NOR channel described later—driven by two example inputs, showing the intermediate stage calculation $T(X)$ and the ϵ-machine of the output process $Y = T(X)$. **b** When driven by an input process consisting of all 1s, the statistical complexity vanishes: $C_X(T) = \log_2(\mathbb{P}(\sigma_0) = 1) = 0$ bits. **c** When driven by the Fair Coin Process, the distribution over causal states is uniform and there is a positive statistical complexity: $C_X(T) = \mathrm{H}(1/2) = 1$ bit. **d** Statistical complexity C_X as a function of the bias of an input Biased Coin Process

Recall that the ϵ-map takes a distribution over joint histories to a distribution over the ϵ-transducer's causal states and that a particular input history defines a distribution over a channel's output history. It follows that a particular input history defines a distribution over joint histories and, therefore, also defines a distribution over an ϵ-transducer's causal states via the ϵ-map. We call this distribution τ:

$$\begin{aligned}\tau(i) &= \mathbb{P}_X\big(S_0 = \sigma_i \,\big|\, \overleftarrow{X}_0 = \overleftarrow{x}\big) \\ &= \mathbb{P}_X\big(\epsilon\big(\overleftarrow{(X,Y)}_0\big) = \sigma_i \,\big|\, \overleftarrow{X}_0 = \overleftarrow{x}\big).\end{aligned}$$

Note that while the ϵ-map takes a particular *joint* history to a unique state, the distribution over states induced by a particular *input* history need not be concentrated on a single state.

When the input process is known and stationary, the ϵ-transducer's stationary distribution π_X can be determined. This provides a starting distribution for the ϵ-transducer, which is updated by the ϵ-transducer's symbol transition matrices as each input symbol is observed and each output symbol is generated. We can therefore calculate the final state distribution $\mathbb{P}(S_L = \sigma_m | x_{0:L}, S_0 \sim \pi_X)$ that results from starting states in distribution π_X and observing finite input word $x_{0:L}$:

$$\begin{aligned}&\mathbb{P}(S_L = \sigma_m | x_{0:L}, S_0 \sim \pi_X) \\ &\quad = \sum_{y_{0:L}} \sum_{i,j,k,\cdots,l} \pi_X(i) T_{ij}^{(y_0|x_0)} T_{jk}^{(y_1|x_1)} \cdots T_{lm}^{(y_{L-1}|x_{L-1})}.\end{aligned}$$

We then obtain τ from the ϵ-transducer by shifting this distribution by L and taking the limit as $L \to \infty$:

$$\begin{aligned}\tau(i) &= \mathbb{P}_X\big(S_0 = \sigma_i \,\big|\, \overleftarrow{X}_0 = \overleftarrow{x}\big) \\ &= \lim_{L\to\infty} \mathbb{P}\big(S_0 = \sigma_i \,\big|\, x_{-L:0}, S_{-L} \sim \pi_X\big).\end{aligned}$$

We can now establish that the ϵ-transducer is an exact presentation of the causal channel that it models.

Proposition 2 (Presentation) *A causal channel's* ϵ-transducer *exactly (and only) reproduces the channel's conditional word probabilities* $\mathbb{P}(Y_{0:L} | \overleftarrow{X}_L)$.

Proof Recall that a *causal* channel's output words do not depend on any inputs occurring *after* the output word. Therefore, the ϵ-transducer must reproduce all of a channel's conditional word probabilities of the form $\mathbb{P}(y_{0:L} | \overleftarrow{x}_L)$. As discussed above, an input history induces a distribution τ over the ϵ-transducer's causal states. So, we calculate the word probabilities directly via repeated application of the ϵ-transducer's symbol transition matrices:

$$\mathbb{P}(y_{0:L} | \overleftarrow{x}_L) = \sum_{i,j,k,\cdots,l,m} \tau(i) T_{ij}^{(y_0|x_0)} T_{jk}^{(y_1|x_1)} \cdots T_{lm}^{(y_{L-1}|x_{L-1})}. \tag{10}$$

□

In fact, we can transduce a *finite* input word even when the channel's behavior depends upon arbitrarily long input histories. By simply starting the ϵ-transducer in its stationary distribution π_X, we have:

$$\mathbb{P}(y_{0:L} | x_{0:L}) = \sum_{i,j,k,\cdots,l,m} \pi_X(i) T_{ij}^{(y_0|x_0)} T_{jk}^{(y_1|x_1)} \cdots T_{lm}^{(y_{L-1}|x_{L-1})}.$$

In either case, we can multiply these conditional word probabilities by the input's word probabilities to obtain joint word probabilities:

$$\mathbb{P}((x, y)_{0:L}) = \mathbb{P}(y_{0:L}|x_{0:L})\mathbb{P}(x_{0:L}).$$

Summing over input words then gives output word probabilities:

$$\mathbb{P}(y_{0:L}) = \sum_{x_{0:L} \in \boldsymbol{\mathcal{X}}^L} \mathbb{P}((x, y)_{0:L}).$$

We can also start the ϵ-transducer with other, arbitrary state distributions when certain initial behavior is desired or if the current internal configuration of the ϵ-transducer is known. This can be very useful in practice, but the resulting generated behavior is no longer guaranteed to be stationary or to match the original channel's behavior. One use of arbitrary state distributions is *real-time* transduction of symbols, where a state distribution ν is repeatedly updated each time-step after a single input symbol x_t is transduced to an output symbol y_t:

$$\mathbb{P}\big(S_{t+1} = \sigma_i \big| (x, y)_t, S_t \sim \nu\big) = \sum_{i,j} \nu(i) T_{ij}^{(y_t|x_t)}.$$

9 Optimality

We now establish that the ϵ-transducer is a channel's unique, maximally predictive, minimal statistical complexity unifilar presentation. Other properties, analogous to those of the ϵ-machine, are also developed. Several proofs parallel those in Refs. [18,19], but are extended from ϵ-machines to the ϵ-transducer. For this initial development, we also adopt a caveat from there concerning the use of infinite pasts and futures. For example, the semi-infinite pasts' entropy $\mathrm{H}[\overleftarrow{Y}]$ is typically infinite. And so, to properly use such quantities, one first introduces finite-length chains (e.g., $\mathrm{H}[Y_{0:L}]$) and at the end of an argument one takes infinite-length limits, as appropriate. Here, as previously, we do not include these extra steps, unless there is subtlety that requires attention using finite-length chains.

Proposition 3 (Causal States Proxy the Past) *When conditioned on causal states, the future output given input is independent of past input and past output:*

$$\mathbb{P}\big(\overrightarrow{Y}_0 \big| \overrightarrow{X}_0, \overleftarrow{(X, Y)}_0, S_0\big) = \mathbb{P}\big(\overrightarrow{Y}_0 \big| \overrightarrow{X}_0, S_0\big).$$

Proof By construction, the causal states have the same future morphs as their corresponding pasts:

$$\mathbb{P}\big(\overrightarrow{Y}_0 \big| \overrightarrow{X}_0, \overleftarrow{(X, Y)}_0\big) = \mathbb{P}\big(\overrightarrow{Y}_0 \big| \overrightarrow{X}_0, S_0\big).$$

Since the causal states are a function of the past—$S_0 = \epsilon\,(\overleftarrow{(X, Y)}_0)$—we also have that:

$$\mathbb{P}\big(\overrightarrow{Y}_0 \big| \overrightarrow{X}_0, \overleftarrow{(X, Y)}_0, S_0\big) = \mathbb{P}\big(\overrightarrow{Y}_0 \big| \overrightarrow{X}_0, \overleftarrow{(X, Y)}_0\big).$$

Combining these two equalities gives the result. □

In other words, when predicting a channel's future behavior from its past behavior, it suffices to use the causal states instead.

Proposition 4 (Causal Shielding) *Past output $\overleftarrow{Y_0}$ and future output $\overrightarrow{Y_0}$ given future input $\overrightarrow{X_0}$ are independent given the current causal state S_0:*

$$\mathbb{P}\big(\overleftrightarrow{Y_0}\big|\overrightarrow{X_0}, S_0\big) = \mathbb{P}\big(\overleftarrow{Y_0}\big|\overrightarrow{X_0}, S_0\big)\mathbb{P}\big(\overrightarrow{Y_0}\big|\overrightarrow{X_0}, S_0\big).$$

Proof We directly calculate:

$$\begin{aligned}\mathbb{P}\big(\overleftrightarrow{Y_0}\big|\overrightarrow{X_0}, S_0\big) &= \mathbb{P}\big(\overleftarrow{Y_0}\big|\overrightarrow{X_0}, S_0\big)\mathbb{P}\big(\overrightarrow{Y_0}\big|\overrightarrow{X_0}, \overleftarrow{Y}, S_0\big)\\ &= \mathbb{P}\big(\overleftarrow{Y_0}\big|\overrightarrow{X_0}, S_0\big)\mathbb{P}\big(\overrightarrow{Y_0}|\overrightarrow{X_0}, S_0\big).\end{aligned}$$

Where the second equality follows from applying Prop. 3 to the second factor. □

In the following, depending on use, we refer to either of the previous propositions as *causal shielding*.

Proposition 5 (Joint Unifilarity) *The current causal state S_0 and current input–output symbol pair $(X, Y)_0$ uniquely determine the next causal state. In this case:*

$$\mathrm{H}[S_1|(X, Y)_0, S_0] = 0.$$

Proof If two pasts are causally equivalent, then either (i) appending a new symbol pair (x, y) to both pasts results in two *new* pasts that are also causally equivalent or (ii) such a symbol pair is never observed when in S_0. We must show that the two new pasts have the same future morph:

$$\overleftarrow{(x, y)} \sim_\epsilon \overleftarrow{(x, y)}' \implies \mathbb{P}\big(\overrightarrow{Y_1}\big|\overrightarrow{X_1}, \overleftarrow{(x, y)}(a, b)\big) = \mathbb{P}\big(\overrightarrow{Y_1}\big|\overrightarrow{X_1}, \overleftarrow{(x, y)}'(a, b)\big),$$

where we have $a \in \boldsymbol{\mathcal{X}}$ and $b \in \boldsymbol{\mathcal{Y}}$ and $\overleftarrow{(x, y)}(a, b) = (\overleftarrow{x}a, \overleftarrow{y}b)$, and the futures $\overrightarrow{Y_1}$ and $\overrightarrow{X_1}$ denote those immediately following the associated conditioning pasts $\overleftarrow{(x, y)}(a, b)$ and $\overleftarrow{(x, y)}'(a, b)$, respectively. Or, we must show that the input–output pair (x, y) is forbidden.

First, let $\overleftarrow{(x, y)} \sim_\epsilon \overleftarrow{(x, y)}'$. Since causal equivalence applies for *any* joint future, it applies to the particular future beginning with symbol pair (a, b):

$$\mathbb{P}\big(b\overrightarrow{Y_1}\big|a\overrightarrow{X_1}, \overleftarrow{(x, y)}\big) = \mathbb{P}\big(b\overrightarrow{Y_1}\big|a\overrightarrow{X_1}, \overleftarrow{(x, y)}'\big).$$

Factoring:

$$\mathbb{P}\big(b\overrightarrow{Y_1}\big|\cdot\big) = \mathbb{P}\big(\overrightarrow{Y_1}\big|Y_0 = b, \cdot\big)\mathbb{P}\big(Y_0 = b\big|\cdot\big)$$

gives:

$$\begin{aligned}&\mathbb{P}\big(\overrightarrow{Y_1}\big|Y_0 = b, a\overrightarrow{X_1}, \overleftarrow{(x, y)}\big)\mathbb{P}\big(Y_0 = b\big|a\overrightarrow{X_1}, \overleftarrow{(x, y)}\big)\\ &\quad= \mathbb{P}\big(\overrightarrow{Y_1}\big|Y_0 = b, a\overrightarrow{X_1}, \overleftarrow{(x, y)}'\big)\mathbb{P}\big(Y_0 = b\big|a\overrightarrow{X_1}, \overleftarrow{(x, y)}'\big).\end{aligned}$$

The second factors on both sides are equal by causal equivalence. So, there are two cases: These factors either vanish or they do not. If they are positive, then we have:

$$\mathbb{P}\big(\overrightarrow{Y_1}\big|Y_0 = b, a\overrightarrow{X_1}, \overleftarrow{(x, y)}\big) = \mathbb{P}\big(\overrightarrow{Y_1}\big|Y_0 = b, a\overrightarrow{X_1}, \overleftarrow{(x, y)}'\big).$$

Rewriting the conditional variables with the symbol pair (a, b) attached to the joint past then gives the first part of the result:

$$\mathbb{P}\big(\overrightarrow{Y_1}\big|\overrightarrow{X_1}, \overleftarrow{(x, y)}(a, b)\big) = \mathbb{P}\big(\overrightarrow{Y_1}\big|\overrightarrow{X_1}, \overleftarrow{(x, y)}'(a, b)\big).$$

In the other case, when the factors vanish, we have:

$$\mathbb{P}\big(Y_0 = b \big| a \overrightarrow{X}_1, \overleftarrow{(x, y)}\big) = \mathbb{P}\big(Y_0 = b \big| a \overrightarrow{X}_1, \overleftarrow{(x, y)}'\big) = 0.$$

This implies that:

$$\begin{aligned}\mathbb{P}\big(Y_0 = b \big| X_0 = a, \overleftarrow{(x, y)}\big) &= \mathbb{P}\big(Y_0 = b \big| X_0 = a, \overleftarrow{(x, y)}'\big)\\ &= 0.\end{aligned}$$

In other words, $Y_0 = b$ is never observed following either past, given $X_0 = a$. That is, (a, b) is forbidden.

It then follows that:

$$\mathrm{H}[S_1 | S_0, (X, Y)_0] = 0,$$

which is equivalent to joint unifilarity when there is a finite number of causal states. □

Unifilarity guarantees that once we know the process is in a particular causal state—we are "synchronized" [32]—we do not lose synchronization over time. This is an important property when using causal states to simulate or predict a system's behavior. Using presentations that are nonunifilar, typically it is necessary to keep track of a *distribution* over states.

Unifilarity is also a useful property to have when attempting to infer an ϵ-transducer from data. Inference of nonunifilar transducers can be challenging, partly due to the existence of multiple possible state paths given a particular start state. Unifilar transducers avoid this problem, effectively reducing the difficulty to that of inferring a Markov chain from data [36]. Finally, unifilarity plays a key role, as a sequel shows, in calculating channel information quantities.

The next theorem shows that ϵ-transducers are input-dependent hidden Markov models.

Proposition 6 (Markovity) *A channel's causal states satisfy the conditional Markov property:*

$$\mathbb{P}\big(S_t \big| X_{t-1}, \overleftarrow{S_t}\big) = \mathbb{P}(S_t | X_{t-1}, S_{t-1}).$$

Proof Since the causal-state transitions are unifilar, there is a well defined set of output symbols $\mathcal{Z} \subseteq \mathcal{Y}$ that causes a transition from state σ_j to state σ_k. We therefore have:

$$\begin{aligned}&\mathbb{P}\big(S_t = \sigma_k \big| X_{t-1}, S_{t-1} = \sigma_j, \overleftarrow{S}_{t-1}\big)\\ &\quad = \mathbb{P}\big(Y_{t-1} \in \mathcal{Z} \big| X_{t-1}, S_{t-1} = \sigma_j, \overleftarrow{S}_{t-1}\big).\end{aligned}$$

Causal shielding applies to finite futures as well as infinite. This, combined with the observation that $\overleftarrow{S}_{t-1}$ is purely a function of the past, allows us to use S_{t-1} to causally shield Y_{t-1} from $\overleftarrow{S}_{t-1}$, giving:

$$\begin{aligned}&\mathbb{P}\big(Y_{t-1} \in \mathcal{Z} \big| X_{t-1}, S_{t-1} = \sigma_j, \overleftarrow{S}_{t-1}\big)\\ &\quad = \mathbb{P}(Y_{t-1} \in \mathcal{Z} | X_{t-1}, S_{t-1} = \sigma_j)\\ &\quad = \mathbb{P}(S_t = \sigma_k | X_{t-1}, S_{t-1} = \sigma_j).\end{aligned}$$

The final equality is again possible due to unifilarity. □

The following theorem shows that the causal states store as much information as possible (from the past) about a channel's future behavior—a desirable property for any predictive model.

Definition 9 The *prescience* of a set $\boldsymbol{\mathcal{R}}$ of rival states—equivalence classes of an alternative partition of pasts—reflects how well the rival states predict a channel's future behavior. Quantitatively, this is monitored by the amount of information they share with future output, given future input:

$$\mathrm{I}\left[R \,;\, \overrightarrow{Y} \middle| \overrightarrow{X}\right],$$

where R is the associated rival-state random variable.

Note that it is sometimes simpler to prove statements about conditional entropy than it is for mutual information. Due to this, we will transform statements about prescience into statements about uncertainty in prediction in several proofs that follow. Specifically, we will make use of the identity:

$$\begin{aligned}\mathrm{I}\left[R_0 \,;\, \overrightarrow{Y}_0 \middle| \overrightarrow{X}_0\right] &= \lim_{L\to\infty} \mathrm{I}\left[R_0 \,;\, Y_{0:L} \middle| \overrightarrow{X}_0\right] \\ &= \lim_{L\to\infty} \left(\mathrm{H}\left[Y_{0:L} \middle| \overrightarrow{X}_0\right] - \mathrm{H}\left[Y_{0:L} \middle| \overrightarrow{X}_0, R_0\right]\right),\end{aligned}$$

where $\mathrm{H}\left[Y_{0:L} \middle| \overrightarrow{X}_0, R_0\right]$ is the finite-future *prediction uncertainty*. Note that the infinite-future prediction uncertainty $\mathrm{H}\left[\overrightarrow{Y}_0 \middle| \overrightarrow{X}_0, R_0\right]$ typically will be infinite, but rewriting the prescience in terms of the limit of finite-future prediction uncertainty allows us to continue to work with finite quantities.

Theorem 1 (Maximal Prescience) *Among all rival partitions $\boldsymbol{\mathcal{R}}$ of joint pasts, the causal states have maximal prescience and they are as prescient as pasts:*

$$\begin{aligned}\mathrm{I}\left[\mathcal{S} \,;\, \overrightarrow{Y} \middle| \overrightarrow{X}\right] &= \mathrm{I}\left[\overleftarrow{(X,Y)} \,;\, \overrightarrow{Y} \middle| \overrightarrow{X}\right] \\ &\geq \mathrm{I}[R \,;\, \overrightarrow{Y} \,|\, \overrightarrow{X}].\end{aligned}$$

Proof We will prove the equivalent statement that the causal states *minimize* finite-future prediction *uncertainty* for futures of any length L and have the same finite-future prediction uncertainty as pasts; i.e., that for all L:

$$\begin{aligned}\mathrm{H}\left[Y_{0:L} \middle| \overrightarrow{X}_0, \mathcal{S}_0\right] &= \mathrm{H}\left[Y_{0:L} \middle| \overrightarrow{X}_0, \overleftarrow{(X,Y)}_0\right] \\ &\leq \mathrm{H}\left[Y_{0:L} \middle| \overrightarrow{X}_0, R_0\right],\end{aligned}$$

Like the causal states, rival states—equivalence classes of an alternative partition $\boldsymbol{\mathcal{R}}$ of pasts—are a function of the past:

$$R = \eta\left(\overleftarrow{(X,Y)}\right).$$

By the Data Processing Inequality [12], we have:

$$\mathrm{H}\left[Y_{0:L} \middle| \overrightarrow{X}_0, \overleftarrow{(X,Y)}_0\right] \leq \mathrm{H}\left[Y_{0:L} \middle| \overrightarrow{X}_0, R_0\right].$$

The causal states share future morphs with their corresponding pasts. By simple marginalization, the same is true for finite-future morphs:

$$\mathbb{P}\left(Y_{0:L} \middle| \overrightarrow{X}_0, \mathcal{S}_0\right) = \mathbb{P}\left(Y_{0:L} \middle| \overrightarrow{X}_0, \overleftarrow{(X,Y)}_0\right) \implies \mathrm{H}\left[Y_{0:L} \middle| \overrightarrow{X}_0, \mathcal{S}_0\right] = \mathrm{H}\left[Y_{0:L} \middle| \overrightarrow{X}_0, \overleftarrow{(X,Y)}_0\right].$$

□

Note that the causal equivalence relation can also be applied directly to a channel that anticipates its future inputs, but the resulting ϵ-transducer has outputs that depend not only upon the current state and input symbol, but some set of *future* input symbols. If the set is finite, then one uses the previous construction that transforms finite anticipation into additional transducer memory (statistical complexity). In this case, the causal states still capture all of the information from the past needed for prediction. That is, they have maximal prescience. Any additional future input dependence, though, must be encoded in the machine's transitions.

Definition 10 A *prescient rival* is an indexed set $\widehat{\mathcal{R}}$ of states (with elements $\widehat{\rho}_i$ and random variable $\widehat{R}$) that is as predictive as any past:

$$\mathrm{I}\left[\widehat{R}\,;\,\overrightarrow{Y}\middle|\overrightarrow{X}\right]=\mathrm{I}\left[\overleftarrow{(X,Y)}\,;\,\overrightarrow{Y}\middle|\overrightarrow{X}\right].$$

Lemma 1 (Refinement) *The partition of a prescient rival $\widehat{\mathcal{R}}$ is a refinement (almost everywhere) of the causal-state partition of the joint input–output pasts.*

Proof Since the causal states' future morphs include every possible future morph of a channel, we can always express a prescient rival's future morph as a (convex) combination of the causal states' future morphs. This allows us to rewrite the entropy over a prescient rival (finite-length) future morph as:

$$\begin{aligned}\mathrm{H}\left[Y_{0:L}\middle|\overrightarrow{x},\widehat{\rho}_k\right]&=\mathrm{H}\left[\mathbb{P}\left(Y_{0:L}\middle|\overrightarrow{x},\widehat{\rho}_k\right)\right]\\&=\mathrm{H}\left[\sum_j\mathbb{P}\left(Y_{0:L}\middle|\overrightarrow{x},\sigma_j\right)\mathbb{P}(\sigma_j|\widehat{\rho}_k)\right].\end{aligned}\tag{11}$$

Since entropy is convex, we also have:

$$\mathrm{H}\left[\sum_j\mathbb{P}\left(Y_{0:L}\middle|\overrightarrow{x},\sigma_j\right)\mathbb{P}(\sigma_j|\widehat{\rho}_k)\right]\geq\sum_j\mathbb{P}(\sigma_j|\widehat{\rho}_k)\,\mathrm{H}\left[Y_{0:L}\middle|\overrightarrow{x},\sigma_j\right].\tag{12}$$

Therefore:

$$\begin{aligned}\mathrm{H}\left[Y_{0:L}\middle|\overrightarrow{X}_0,\widehat{R}_0\right]&=\sum_k\mathbb{P}(\widehat{\rho}_k)\,\mathrm{H}\left[Y_{0:L}\middle|\overrightarrow{X}_0,\widehat{\rho}_k\right]\\&\geq\sum_k\mathbb{P}(\widehat{\rho}_k)\sum_j\mathbb{P}(\sigma_j|\widehat{\rho}_k)\,\mathrm{H}\left[Y_{0:L}\middle|\overrightarrow{X}_0,\sigma_j\right]\\&=\sum_{j,k}\mathbb{P}(\sigma_j,\widehat{\rho}_k)\,\mathrm{H}\left[Y_{0:L}\middle|\overrightarrow{X}_0,\sigma_j\right]\\&=\sum_j\mathbb{P}(\sigma_j)\,\mathrm{H}\left[Y_{0:L}\middle|\overrightarrow{X}_0,\sigma_j\right]\\&=\mathrm{H}\left[Y_{0:L}\middle|\overrightarrow{X}_0,S_0\right],\end{aligned}$$

where the inequality follows from Eqs. (11) and (12). Since the rival states $\widehat{R}$ are prescient, we know that equality must be attained in this inequality for each L. Equality is only possible when $\mathbb{P}(\sigma_j|\widehat{\rho}_k)=1$ for exactly one value of j and vanishes for every other j. That is, if a rival state is prescient, it is contained entirely within a single causal state, aside from a set of measure zero. Thus, the partition of the prescient rival states is a refinement of the causal-state partition almost everywhere. □

Theorem 2 (Minimality) *For any given input process X, causal states have the minimal conditional statistical complexity among all prescient rival partitions $\widehat{R}$:*

$$C_X(S) \le C_X\left(\widehat{R}\right).$$

Proof Since a prescient rival partition is a refinement almost everywhere, there exists a function f defined almost everywhere that maps each rival state to the causal state that (almost everywhere) contains it:

$$f(\widehat{\rho}_i) = \sigma_j.$$

Then, we have:

$$\begin{aligned} \mathrm{H}_X\left[\widehat{R}\right] &\ge \mathrm{H}_X\left[f\left(\widehat{R}\right)\right] \\ &= \mathrm{H}_X[S]. \end{aligned}$$

□

Corollary 1 *Causal states minimize the channel complexity $\overline{C_\mu}$.*

Proof Immediate from the preceding theorem. □

In words, we established the fact that the causal states store all of the information contained in the past that is necessary for predicting a channel's future behavior and as little of the remaining information "overhead" contained in the past as possible. Given an input process, ϵ-transducer causal states maximize $\mathrm{I}[\overrightarrow{Y}; S|\overrightarrow{X}]$ while minimizing $\mathrm{I}[\overleftarrow{(X,Y)}; S]$.

The final optimality theorem shows that any states which have these properties are in fact the causal states.

Theorem 3 (Uniqueness) *The ϵ-transducer is the unique prescient, minimal partition of pasts. If $C_X\left(\widehat{R}\right) = C_X(S)$ for every input process $\overleftrightarrow{X}$, then the corresponding states $\widehat{R}$ and S are isomorphic to one another almost everywhere. And, their equivalence relations $\sim_\eta$ and $\sim_\epsilon$ are the same almost everywhere.*

Proof Again, the Refinement Lemma (Lemma 1) says that $S = f\left(\widehat{R}\right)$ almost everywhere. It therefore follows that $\mathrm{H}_X\left[S|\widehat{R}\right] = 0$. Moreover, by assumption $\mathrm{H}_X[S] = \mathrm{H}_X\left[\widehat{R}\right]$. Combining these with the symmetry of mutual information gives:

$$\begin{aligned} \mathrm{I}_X\left[S;\widehat{R}\right] &= \mathrm{I}_X\left[\widehat{R};S\right] \\ \mathrm{H}_X[S] - \mathrm{H}_X\left[S|\widehat{R}\right] &= \mathrm{H}_X\left[\widehat{R}\right] - \mathrm{H}_X\left[\widehat{R}|S\right] \\ \mathrm{H}_X[S] - 0 &= \mathrm{H}_X[S] - \mathrm{H}_X\left[\widehat{R}|S\right] \\ \mathrm{H}_X\left[\widehat{R}|S\right] &= 0. \end{aligned}$$

The latter holds if and only if there is a function g such that $\widehat{R} = g(S)$ almost everywhere. By construction g is the inverse f^{-1} of f almost everywhere. We have that $f \circ \eta = \epsilon$ and $f^{-1} \circ \epsilon = \eta$.

Finally, the two equivalence relations $\sim_\epsilon$ and $\sim_\eta$ are the same almost everywhere:

$$\begin{aligned}
&\overleftarrow{(x, y)} \sim_\epsilon \overleftarrow{(x, y)}' \\
\implies &\epsilon\left(\overleftarrow{(x, y)}\right) = \epsilon\left(\overleftarrow{(x, y)}'\right) \\
\implies &f^{-1} \circ \epsilon\left(\overleftarrow{(x, y)}\right) = f^{-1} \circ \epsilon\left(\overleftarrow{(x, y)}'\right) \\
\implies &\eta\left(\overleftarrow{(x, y)}\right) = \eta\left(\overleftarrow{(x, y)}'\right) \\
\implies &\overleftarrow{(x, y)} \sim_\eta \overleftarrow{(x, y)}',
\end{aligned}$$

and

$$\begin{aligned}
&\overleftarrow{(x, y)} \sim_\eta \overleftarrow{(x, y)}' \\
\implies &\eta\left(\overleftarrow{(x, y)}\right) = \eta\left(\overleftarrow{(x, y)}'\right) \\
\implies &f \circ \eta\left(\overleftarrow{(x, y)}\right) = f \circ \eta\left(\overleftarrow{(x, y)}'\right) \\
\implies &\epsilon\left(\overleftarrow{(x, y)}\right) = \epsilon\left(\overleftarrow{(x, y)}'\right) \\
\implies &\overleftarrow{(x, y)} \sim_\epsilon \overleftarrow{(x, y)}'.
\end{aligned}$$

□

ϵ-Transducer uniqueness means that C_X is *the* conditional complexity of a channel and, therefore, justifies calling $\overline{C_\mu}$ *the* channel complexity.

10 Global ϵ-Machine Versus ϵ-Transducer

Given a particular joint process or its global ϵ-machine, it is possible (provided that the input process satisfies certain requirements) to construct the ϵ-transducer that maps input $\overleftrightarrow{X}$ to output $\overleftrightarrow{Y}$, such that $\overleftrightarrow{(X, Y)} = \left(\overleftrightarrow{X}, f(\overleftrightarrow{X})\right)$, where f is the transducer and $f\left(\overleftrightarrow{X}\right)$ designates the output of the transducer, given input process $\overleftrightarrow{X}$. Sequels address the relationship between a joint process' global ϵ-machine and corresponding ϵ-transducer at both the process (channel) level and at the automata (ϵ-machine and ϵ-transducer) level. There, we provide algorithms for "conditionalizing" a joint process or ϵ-machine to obtain the corresponding channel or ϵ-transducer, as well as algorithms for obtaining input or output marginals, applying an ϵ-transducer to an input ϵ-machine, composing multiple ϵ-transducers, and inverting an invertible ϵ-transducer.

Note that the ability to construct an ϵ-transducer from a joint process can be useful when attempting to infer an ϵ-transducer from data, as such data will typically come from a system driven by some *particular* (possibly controllable) input; i.e., the data is a sample of a *joint process*.

11 History ϵ-Transducer Versus Generator ϵ-Transducer

The preceding focused on the *history* specification of an ϵ-transducer, where a machine is obtained by partitioning a channel's histories (joint pasts). We can also consider the *generator* specification, where we instead *start* with a machine that produces a stationary, ergodic channel. Taking this perspective, an ϵ-transducer is an input-dependent, strongly connected, aperiodic hidden Markov model with unifilar transitions and probabilistically distinct states. The history and generator specifications of an ϵ-transducer are likely equivalent—as they are with ϵ-machines [21]—but we leave such a proof to future work.

12 Examples Revisited

With the ϵ-transducer defined, we can revisit the example channels examined above. We display channel structure via its ϵ-transducer's state transition diagram, as well as a wordmap that now colors each joint history based on its corresponding causal state. Input and output history projections are also shown. Histories that are not mapped to a single causal state are colored black. Recall that causal states partition *joint* histories, so input or output histories alone need not correspond to a unique causal state. We will discuss Markov orders for these channels, but we leave it to the reader to construct an exhaustive list of Markov orders for each channel.

Since the Identity (Fig. 7), All is Fair (Fig. 8), and Z (Fig. 9) Channels are memoryless, their behavior does not depend on the past. As a result, there is a single causal state containing every past and so their wordmaps are monochromatic. Since they have a single causal state, $C_X = \overline{C_\mu} = 0$.

In contrast, the Delay Channel (Fig. 10) has two causal states and so two colors corresponding to pasts in which the input ends on a 0 (left half of the wordmap) or a 1 (right half of the wordmap). This partitioning into halves is a characteristic of channels with a pure feedforward Markov order $R_{\text{pff}} = 1$. We also see that the output words are colored black, illustrating the fact that the output tells us nothing about the current causal state. The channel's pure feedforward Markov order of 1 can be seen in the channel's ϵ-transducer

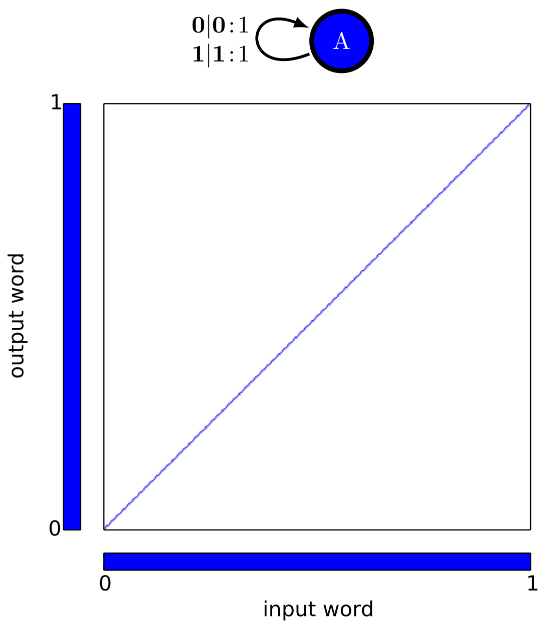


Fig. 7 Identity Channel: ϵ-Transducer and causal-state colored wordmap. See text for explanation

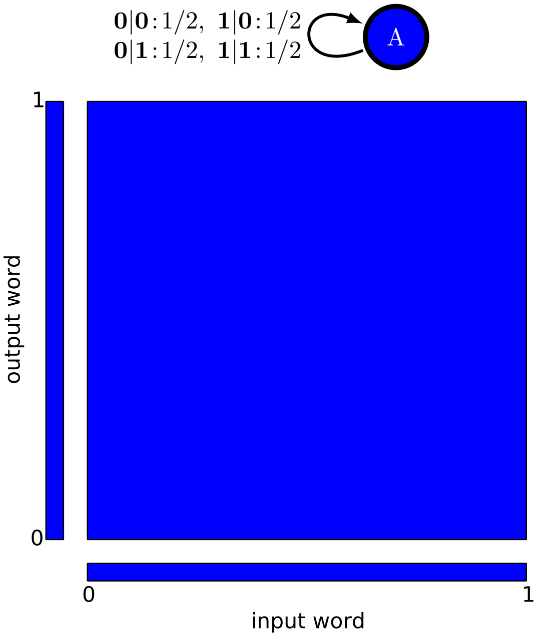


Fig. 8 All-Is-Fair Channel: ϵ-Transducer and causal-state colored wordmap

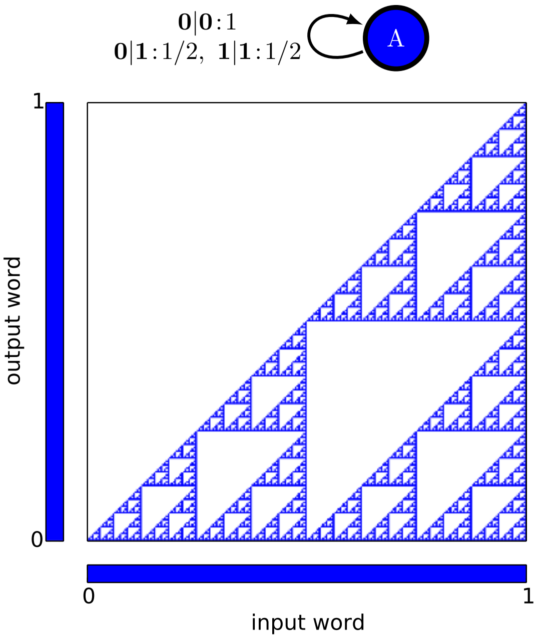


Fig. 9 Z Channel: ϵ-Transducer and causal-state colored wordmap

state-transition diagram by observing that all transitions on input symbol 0 lead to state A and all transitions on input symbol 1 lead to state B. Since the Delay Channel is undefined for outputs alone (R_{pfb} is undefined), it is the first example channel with a nontrivial irreducible feedforward order: $R_{\text{iff}} = 1$. The causal states of the Delay Channel simply store a single bit of input, their entropy therefore matches the length-1 block entropy of the input process:

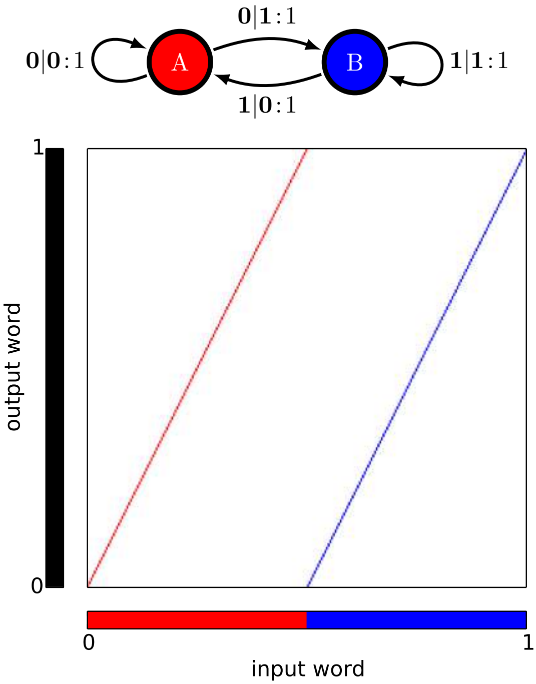


Fig. 10 Delay Channel: ϵ-Transducer and causal-state colored wordmap

$C_X = H[X_0]$. Maximizing the latter over input process gives us $\overline{C_\mu} = 1$, attained with Fair Coin Process input.

The Feedforward NOR Channel (Fig. 11) also stores the previous input symbol and, therefore, partitions input histories the same way ($R_{\text{pff}} = 1$). We see in the wordmap that there is again an ambiguity of causal state given output histories alone and there is, therefore, no pure feedback presentation for the channel. Specifically, we see that the ambiguity arises for histories where the output ends on two 0s (lower quarter of the wordmap). This can be verified in the channel's ϵ-transducer by observing that a 1 on output always leads to state A, but a 0 on output only leads to a unique state if it is followed by a 1 on output. A single symbol of input is *always* needed to guarantee well defined behavior ($R_{\text{iff}} = 1$). Since the Feedforward NOR Channel's causal states store the same information as the Delay Channel, we can again drive the channel with the Fair Coin Process to attain $\overline{C_\mu} = 1$ bit.

The wordmap for the Odd NOT Channel (Fig. 12) has projected input (and output) partitions with structure at all scales. This is the signature of states that depend upon infinite histories—one must provide an arbitrarily long binary expansion to specify the location of the causal state boundaries and, therefore, the causal states themselves. If we observe both inputs and outputs, we only need to specify in which quadrant a joint history lies in order to determine its causal state. That is, the Odd NOT Channel is sofic on both input and output alone (infinite pure feedforward and pure feedback Markov orders, R_{pff} and R_{pfb}, respectively), but Markovian when both input and output are considered (finite Markov order R). We also see that the causal states store the *same information* (parity) about input histories as they do output histories, by observing the symmetry along the diagonal. Since the Period-2 Process generates sequences that always alternate between even and odd parity, we can drive the channel with this process to induce a uniform distribution over its causal states. Therefore, we have $\overline{C_\mu} = 1$ bit, again.

Fig. 11 Feedforward NOR Channel: ϵ-Transducer and causal-state colored wordmap

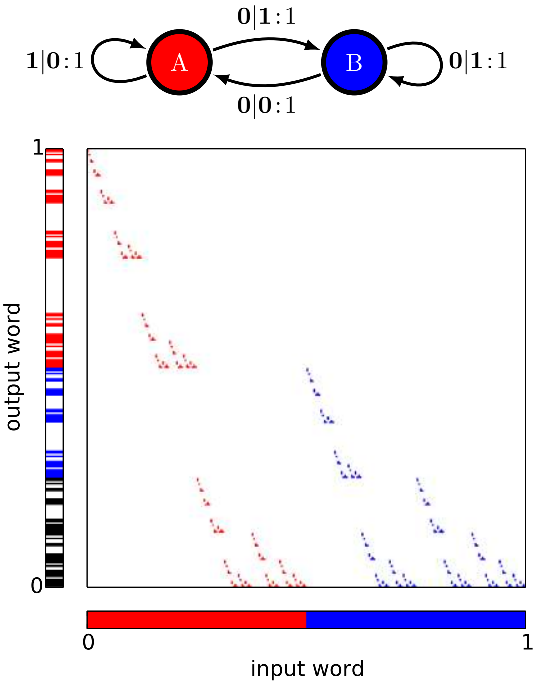


Fig. 12 Odd NOT Channel: ϵ-Transducer and causal-state colored wordmap

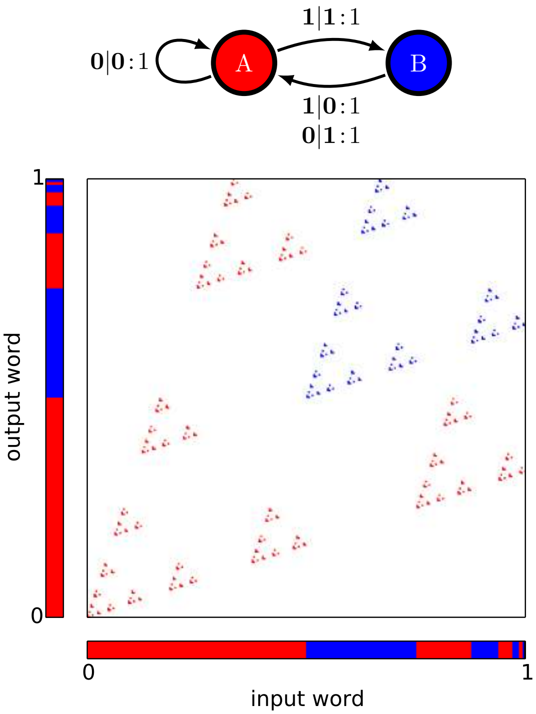
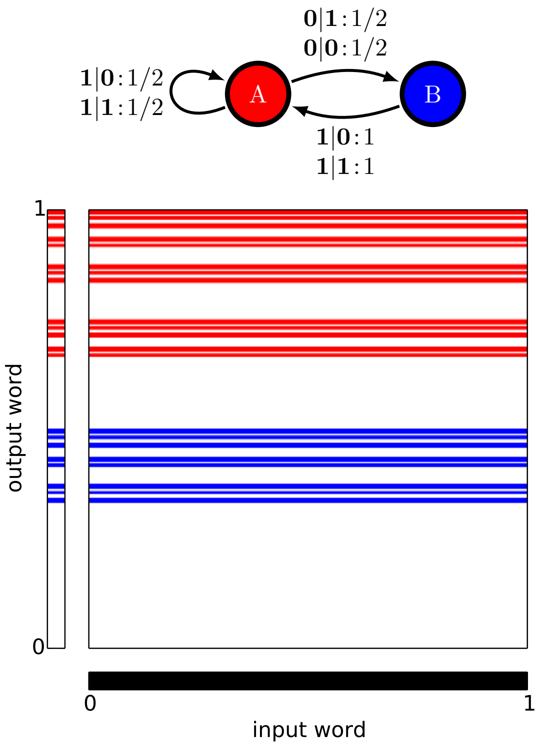


Fig. 13 All-Is-Golden Channel: ϵ-Transducer and causal-state colored wordmap



We see that the All-Is-Golden Channel (Fig. 13) has a pure feedback Markov order of $R_{\text{pfb}} = 1$. Since it ignores its input, however, input histories tell us nothing about in which causal state the channel is. We also see that the wordmap is horizontally symmetric due to this lack of input dependence. Since the state transitions depend only on output, the state distribution and, therefore, the statistical complexity are independent of input. In particular, the channel's statistical complexity is that of the Golden Mean Process (GMP): $C_X = \overline{C_\mu} = C_\mu(\text{GMP}) \approx 0.918$ bits.

The wordmap for the Feedback NOR Channel (Fig. 14) clearly shows that it has infinite pure feedforward Markov order R_{pff}, but finite Markov R and pure feedback Markov R_{pfb} orders. Contrast this with the wordmap for the Feedback XOR Channel (Fig. 15) clearly showing that the causal state, and so the channel's behavior, cannot be determined by input alone. Observe that the Feedback NOR Channel is in state A with probability 1 when a 1 is observed on input and oscillates between states A and B if the channel is driven with a period-2 cycle of 0s and 1s from that point on. We can therefore induce a uniform distribution over causal states by driving the channel with the Period-2 Process. We can also induce a uniform distribution over the Feedback XOR Channel's causal states by driving the channel with the Fair Coin Process, which causes all state transitions to occur with equal probability. In both cases, $\overline{C_\mu} = 1$ bit.

The Odd Random Channel (Fig. 16) has infinite pure feedforward and pure feedback Markov orders ($R_{\text{pff}} = R_{\text{pfb}} = \infty$), but unlike the Odd NOT channel, the Markov order R is infinite. In the Odd NOT channel, we saw structure at all scales in the input and output projections of the wordmap, but a partitioning into quadrants in the complete wordmap. Now, we see that there is no such simple partition in the complete wordmap, and there is structure at

Fig. 14 Feedback NOR Channel: ϵ-Transducer and causal-state colored wordmap

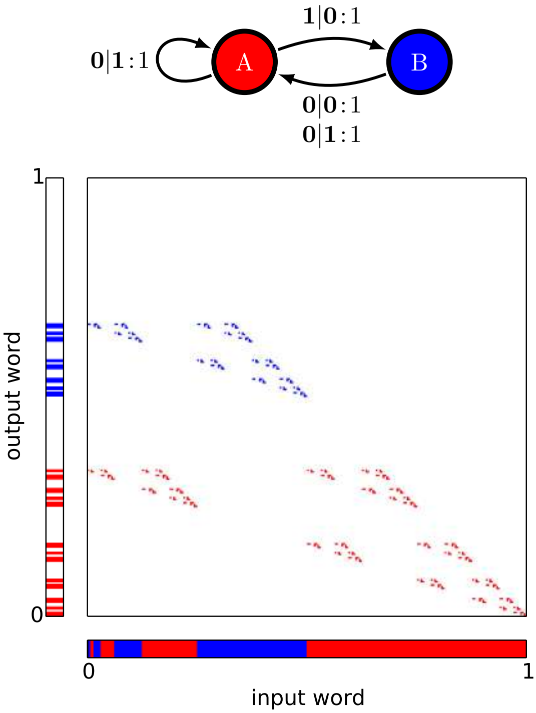
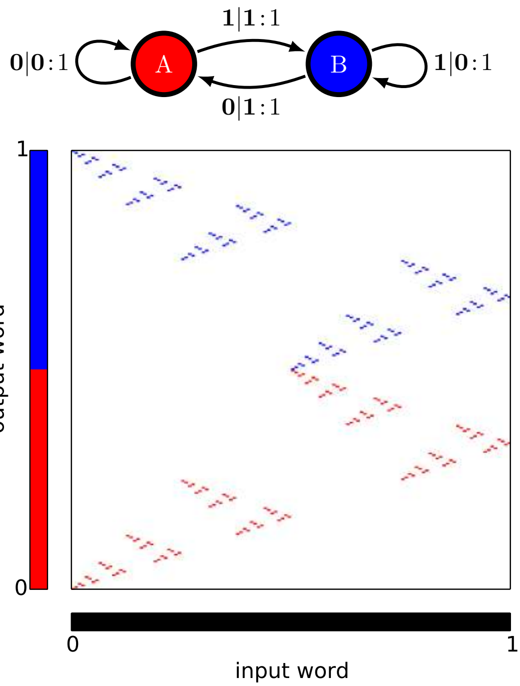


Fig. 15 Feedback XOR Channel: ϵ-Transducer and causal-state colored wordmap



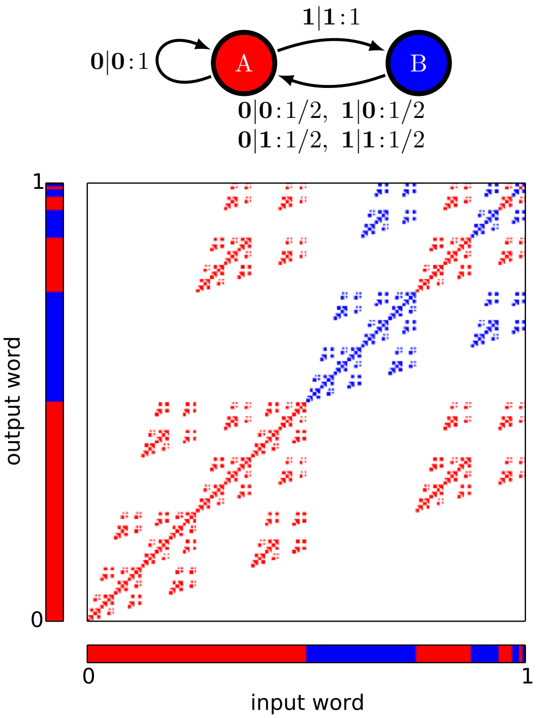


Fig. 16 Odd Random Channel: ϵ-Transducer and causal-state colored wordmap

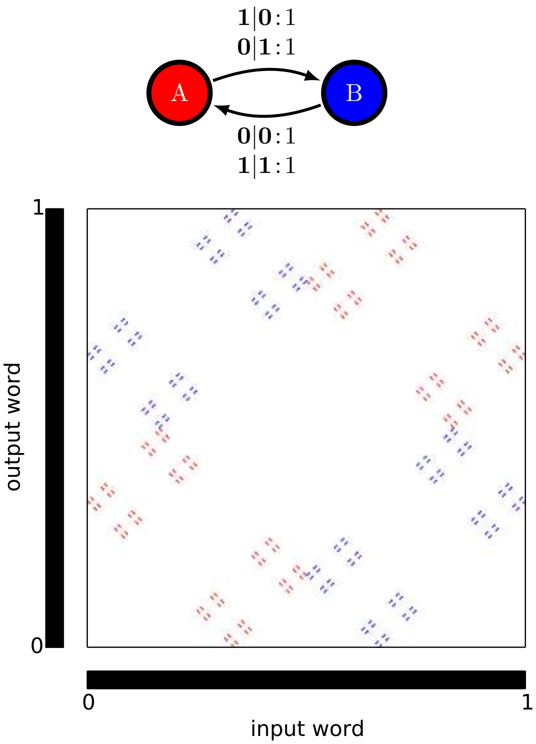


Fig. 17 Period-2 Identity NOT Channel: ϵ-Transducer and causal-state colored wordmap

all scales in the coloring of histories. In other words, one must specify arbitrarily long *pairs* of binary expansions (input and output) in order to identify a causal state. Specifically, knowing that a past ended in $(x, y)_t = (1, 1)$ does not uniquely determine a causal state. This can be seen as multiple colors appearing in the upper-right quadrant of the wordmap. If, however, we know that the *previous* pair was $(0, 0)$, $(0, 1)$, or $(1, 0)$, we see that the history maps to causal state B; corresponding to the upper-left, lower-left, and lower-right subquadrants of the upper-right quadrant, respectively. Similarly, if the previous pair was $(1, 1)$, we are left with an ambiguity in state; corresponding to the upper-right subquadrant of the upper-right quadrant. Therefore, we require an arbitrarily long past to determine the channel's causal state in general. Since the causal states store the same parity as the Odd NOT channel, we can again drive the channel with the Period-2 Process to induce a uniform distribution, giving us $\overline{C_\mu} = 1$ bit.

The Period-2 Identity NOT Channel (Fig. 17) has a Markov order of $R = 1$, but we clearly see that neither input nor output alone determines the channel's causal state ($R_{\text{iff}} = R_{\text{ifb}} = 1$). Since the states have a uniform distribution regardless of input, we have $C_X = \overline{C_\mu} = 1$ bit.

13 Infinite-State ϵ-Transducers: The Simple Nonunifilar Channel

The example channels were chosen to have a finite number of causal states (typically two), largely to keep the analysis of their structure accessible. We can see that even with a few states, ϵ-transducers capture a great deal of behavioral richness. Nonetheless, many channels have an infinite number of causal states. Consider, for example, the *Simple Nonunifilar Channel*. This channel's behavior is captured simply by the finite-state presentation shown in Fig. 18.

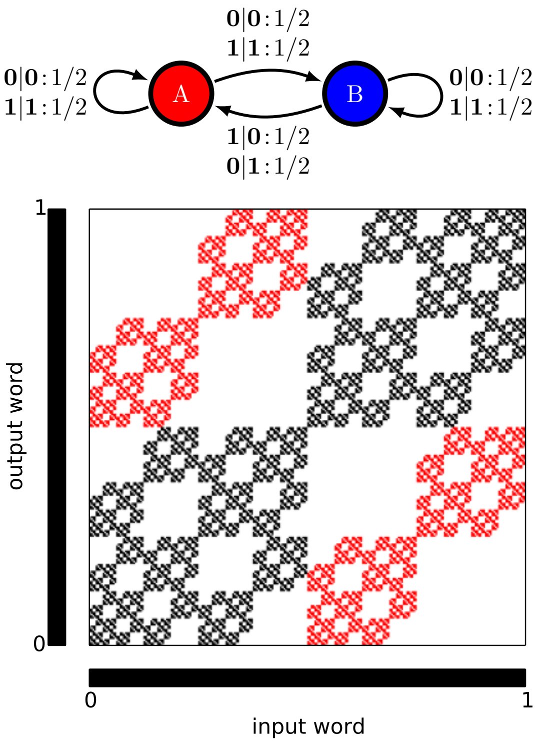


Fig. 18 Simple Nonunifilar Channel: *Nonunifilar* transducer presentation and state colored wordmap

When in state A, the channel behaves as the identity and has an equal probability of staying in state A or transitioning to state B. When in state B, the channel will either behave as the identity and transition back to state B or behave as the bit flipped identity and transition to state A, each with equal probability.

Observe that the transducer shown is *non*unifilar and is, therefore, *not* the ϵ-transducer for the channel. For example, the joint symbol (0, 0) can cause state A to transition to either itself or state B. This nonunifilarity manifests in the wordmap as large blocks of black points. These indicate joint histories that lead to a *mixture* of transducer states. This illustrates the fact that an observer cannot typically retain synchronization to a *particular* state of a nonunifilar transducer—a problem not present when using unifilar transducers.

It is possible to construct the ϵ-transducer for the Simple Nonunifilar Channel, but doing so results in a transducer with a countably infinite set of states. This minimal, unifilar ϵ-transducer can be seen in Fig. 19. Since any channel with a finite Markov order (and finite alphabet) will have a finite number of causal states, a channel with an infinite number of causal states will have infinite Markov order. This is also evident in the causal-state wordmap for the Simple Nonunifilar Channel, as one needs infinite resolution (in general) to determine to which causal state a joint past leads. In fact, even if we know either the infinite input or output past, we still need to know the full output or input past, respectively, in order to characterize the channel's behavior. This is therefore the first example we have seen with $R_{\text{iff}} = R_{\text{ifb}} = \infty$.

Observe that while the Simple Nonunifilar Channel's output clearly depends upon its input, its state-to-state transitions do *not*. Its statistical complexity is therefore independent of the input process chosen. In fact, the causal states and transitions between them are identical to the Simple Nonunifilar Source [22]. The statistical complexity is therefore equal to the statistical complexity of the Simple Nonunifilar source: $C_X = \overline{C_\mu} \approx 2.71$ bits. Even though there are an infinite number of states, the B_i states are occupied with probability that decreases quickly with i, thus allowing for a finite Shannon state entropy. Note that if one were to use Fig. 18's *nonunifilar* presentation for the Simple Nonunifilar Channel, the statistical complexity would be underestimated as $C_X = \overline{C_\mu} = 1$ bit.

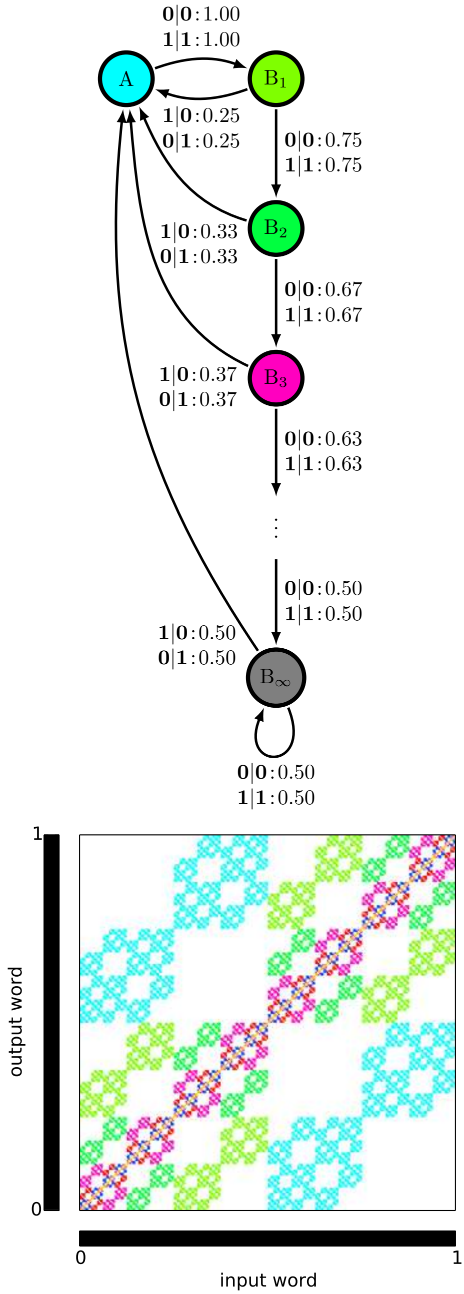
14 Discussion

Previously, we described computational mechanics in the setting of either generating or controlling processes [15]. As noted there, generation and control are complementary. Here, we developed computational mechanics in a way that merges both control (the input process) and generation (the output process), extending the ϵ-machine to the ϵ-transducer. With this laid out, we describe how the ϵ-transducer overlaps and differs from alternatives to modeling input–output processes. We then turn to discuss applications, which incidentally elucidate our original motivations, and suggest future directions.

14.1 Related Work: Modeling

Following the signposts of earlier approaches to modeling complex, nonlinear dynamical systems [2,37], we are ultimately concerned with reconstructing a transducer when given a general channel or given a joint process, either analytically or via statistical inference. And so, when discussing related efforts, we distinguish between those whose goal is to extract a model, which we review now, and those that analyze types of transductions, which we review next. After this, we turn to applications.

Fig. 19 Simple Nonunifilar Channel: ϵ-Transducer and causal-state colored wordmap



For statistical estimation we note that the recently introduced Bayesian Structural Inference (BSI) [36] allows one to estimate the posterior probability that ϵ-machines generate a given, even relatively short, data series. BSI's generality allows it to be readily adapted to infer ϵ-transducers from samples of an input–output process. This turns on either developing an enumeration of ϵ-transducers which parallels that developed for ϵ-machines in Ref. [38] or on developing a list of candidate ϵ-transducers for a given circumstance. And, these are also readily accomplished. A sequel provides the implementations. Previously, inferring causal states, and so causal-state filters, had also been addressed; see, for example, Refs. [36,39,40].

Optimal transducers were originally introduced as structure-based filters to define hierarchical ϵ-machine reconstruction [22,41] in terms of causal-state filtering, to detect emergent spatiotemporal patterns [42–45], and to explore the origins of evolutionary selection pressure [46] and the evolution of language structure [47]. These causal-state transducers were first formalized in Ref. [48] and several of those results are reproduced in Ref. [49]. Appendix shows that the definition there, which makes additional assumptions compared to that here, are equivalent. The more-general development is more elegant, in that it establishes unifilarity, for example, rather than assume such a powerful property. Likely, in addition, the generality will allow ϵ-transducers to be more widely used.

Throwing the net wider—beyond these, most directly related, prior efforts—there have been many approaches to modeling input–output mappings. We will use the fact that most do not focus on quantitatively analyzing the mapping's intrinsic structure to limit the scope of our comments. We mention a few and then only briefly. Hopefully the list, nonetheless, suggests directions for future work in these areas.

Today, many fall under the rubric of learning, though they are rather more accurately described as statistical parameter estimation within a fixed model class. Probably, the most widely used and developed methods to model general input–output mappings are found in artificial neural networks [50,51] and in the more modern approaches that employ kernel methods [52], statistical physics [53], and information theory [54,55]. Often these methods require IID-drawn samples and so do not directly concern mappings from one temporal process to another. Unlike ϵ-transducers, they are also typically limited to model classes—e.g., feedforward and directed acyclic graph structures—that do not allow internal feedback or dynamics.

That said, neural networks that are recurrent are universal approximators of dynamical systems and, per force, are channels with feedback and feedforward memory [56]. They are well known to be hard to train and, in any case, rarely quantitatively analyzed for the structures they capture when successfully trained. In the mathematical statistics of time series, for comparison, AutoRegressive-Moving-Average model with eXogenous inputs model (ARMAX models) are channels with feedback and feedforward memory, but they are *linear*—current output is a linear combination of past inputs and outputs. The nonlinear generalization is the Nonlinear AutoRegressive eXogenous model (NARX), which is a very general memoryful causal channel. At some future time, likely using ϵ-transducers extended to continuous variables as recently done for ϵ-machines in Ref. [57], we will understand better the kinds of structure these channels can represent.

14.2 Related Work: Classification

Beyond developing a theoretical framework for structured transformations, one that is sufficiently constructive to be of use in statistical inference, there are issues that concern how they give a new view, if any, of the organization of the space of structured processes itself.

Specifically, computational mechanics up to this point focused on processes and developed ϵ-machines to describe them as stochastic sets. ϵ-machines are, most simply stated, compact representations of distributions over sequences. With the ϵ-transducers introduced here, computational mechanics now has formalized stochastic mappings of these stochastic sets. And, to get to the point, with sets and mappings one finally has a framework capable of addressing the recoding equivalence notion and the geometry of the space of processes proposed in Ref. [58]. A key component of this will be a measure of distance between processes that uses a structural measure from the minimal optimal mapping (ϵ-transducer) between them. This would offer a constructive, in the sense we use the word, approach to the view of process space originally introduced by Shannon [13,59,60].

This then leads to the historically prior question of structurally classifying processes—paralleling schemes developed in computation theory [24]. Indeed, our development is much closer to input–output processes from the earliest days of dynamical systems and automata theory—which were concerned with exploring the range of behaviors of mechanical systems and the then-new digital computers.

Briefly, ϵ-transducers are probabilistic endomorphisms of subshifts as studied in symbolic dynamics [25]. The (nonprobabilistic) endomorphisms there were developed to explore the equivalence of processes via conjugacies. Notably, this area grew out of efforts in the 1920s and 1930s by Hedlund, Morse, Thue, and others to define symbolic dynamical systems that were more analytically tractable than continuum-state systems [61]. Their efforts played a role in forming the logical foundations of mathematics and so eventually in the emergence of a theory of computation via Church, Gödel, Post, and Turing [62–65]. This led eventually to Moore's abstractions of sequential machines and transducers [66] and to Huffman's concept of a minimal implementation [67] and information lossless automata [68–70]. Today, though expositions are increasingly rare, finite-state transducers are covered by several texts on computation theory; see, for example, Ref. [71].

Once one allows for distributions over sequences, though, then one shifts from the overtly structural approach of symbolic dynamics and automata to Shannon's information sources and communication channels [26] and a strong emphasis on stochastic process theory. As noted in the introduction, one principal difference is that here we considered channels with memory, while the latter in its elementary treatments considers memoryless channels or channels with very restricted forms of memory. Finite-state channels have been developed in limited way, though; for example, see Ref. [72, Ch.7] and for very early efforts see Refs. [73] and [74]. There are also overlaps, as we attempted to show in the selected examples, with classifications developed in digital filter theory [75].

There are also differences in focus and questions. Whereas information theory [12,26] studies *quantities* of information such as intrinsic randomness and informational correlation, computational mechanics [1] goes an additional step and attempts to quantify the information *itself*—the *computational structure* or memory within a system. This is achieved not by assuming a class of model directly, but by making a simple assumption about modeling itself: The only relevant information is that which contributes to prediction—the "difference that makes a difference" to the future [76]. Via the causal equivalence relation, this assumption leads directly to the *unique*, *maximally predictive*, and *minimally complex* model of our measurement data—the ϵ-machine. Another way to express this is that ϵ-transducers give a constructive way to explore the information theory of channels with and without memory.

14.3 Applications

Our development of ϵ-transducers was targeted to provide the foundation for several related problems—problems that we will address elsewhere, but will briefly describe here to emphasize general relevance and also to suggest future directions.

14.3.1 Inference Versus Experimentation

If all data collected is produced by a measuring device, then any model formed from that data captures both the structure of the system *and sensor* in combination. Is there a natural separation of measuring instrument from system measured? We can now pose this precisely in terms of ϵ-transducers: Are there optimal decompositions of a process's ϵ-machine into a possibly smaller ϵ-machine (representing the hidden process) composed with a ϵ-transducer (representing the measuring instrument)?

14.3.2 Information Flow Within and Between Systems

In nonlinear dynamics and in information theory there has been a long-lived interest in how information "flows" and how such flows relate to a system's mechanical organization; see Refs. [77–80], to mention only a few. These employ specializations of Eq. (5)'s excess entropy, being various forms of conditional mutual information. The *transfer entropy* [81] and the earlier *directional information* [82,83] are two such. The main issue concerns how one process affects another and so this is a domain in which ϵ-transducers—as optimal models of the structured transformations between processes—can help clarify the issues.

In particular, there has been recent criticism of the use of these as measures of information flow and, specifically, their relation to the structural organization of the flows [84]. We can now do better, we believe, since ϵ-transducers give a canonical presentation with which to describe and extract the structure of such mappings. And this, in turn, allows one to explicitly relate how causal-state structure supports or precludes information flows. We address this problem in a sequel [85].

14.3.3 Process Decomposition

Given a process, we can now analyze what internal components drive or are driven by other internal components. As one example, Is a subset of the measurement alphabet the "output" being driven by another subset that is "input"? The question hints at the solution that one can now provide: Produce the ϵ-transducer for each bipartite input–output partitioning of the global ϵ-machine alphabet, giving a set of candidate input–output models. One can then invoke, based on a notion of first principles (such as parsimony) or prior knowledge, a way to choose the "best" input–output, driver-drivee decomposition.

14.3.4 Perception-Action Cycles

Probably one of the most vexing contemporary theoretical and practical problems, one that occurs quite broadly, is how to describe long-term and emergent features of *dynamic learning* in which a system models its input, makes a decision based on what it has gleaned, and takes an action that affects the environment producing the inputs. In psychology and cognitive sciences this problem goes under the label of the *perception-action cycle*; in neuroscience, under *sensori-motor loop* [7,8]. The problem transcends both traditional mathematical statistics and modern machine learning, as their stance is that the data is not affected by what is learned. And in this, it transcends the time-worn field of experiment design [86,87] and the more recent machine learning problem of *active learning* [53]. Though related to computational mechanics via Ref. [40], the recent proposal [88] for *interactive learning* is promising, but is not grounded in a systematic approach to structure. It also transcends control theory, as the latter does not address dynamically building models, but rather emphasizes how to monitor and drive a given system into given states [89].

ε-Transducers suggest a way to model the transduction of sensory input to a model and from the model to a decision process that generates actions. Thus, the computational mechanics representation of the perception-action cycle is two cross-coupled ϵ-transducers—one's output is the other's input and vice versa. Formulating the problem in this way promises progress in analyzing and in quantifying structures in the space of models and strategies.

Physical applications of ϵ-transducers to analyze the information thermodynamics of feedback control in Maxwellian Demons can be seen in Szilard's Engine [4] and the Mandal-Jarzynski ratchet [3,5].

15 Conclusion

Previously, computational mechanics focused on extracting and analyzing the informational and structural properties of individual processes. The premise being that once a process's ϵ-machine had been obtained, it can be studied in lieu of other more cumbersome (or even inappropriate) process presentations. Since the ϵ-machine is also unique and minimal for a process, its structure and quantities were treated as being those of the underlying system that generated the process. Strengthening this paradigm, virtually all of a process's correlational, information, and structural quantities can now be calculated in closed form using new methods of ϵ-machine spectral decomposition [90].

By way of explaining this paradigm, we opened with a review of stationary processes and their ϵ-machines, turning to broaden the setting to joint input–output processes and communication channels. We then defined the (conditional) causal equivalence relation, which led immediately to transducer causal states and the ϵ-transducer. A series of theorems then established their optimality. To illustrate the range of possible transformations we considered a systematic set of example channels that, in addition, provided an outline of a structural classification scheme. As an aide in this, we gave a graphical way to view structured transformations via causal-state wordmaps. With the framework developed, one sees that the same level of computational mechanics' prior analysis of individual processes can now be brought to bear on understanding structural transformations between processes.

The foregoing, however, is simply the first in a series on the structural analysis of mappings between processes. The next will address the information-theoretic measures appropriate to joint input–output processes. We then will turn to an analysis that blends the present results on the causal architecture of structured transformations and the information-theoretic measures, showing how the internal mechanism expressed in the ϵ-transducer supports information creation, loss, and manipulation during flow. From that point, the sequels will branch out to address channel composition, decomposition, and inversion.

Given the diversity of domains in which structured transformations (and their understanding) appear to play a role, there looks to be a wide range of applications. In addition to addressing several of these applications, Sect. 14 outlined several future research directions. The ϵ-transducer development leads, for example, to a number of questions that can now be precisely posed and whose answers now seem in reach: How exactly do different measuring devices change the ϵ-machine formed from measurements of a fixed system? What precisely is lost in the measurement process, and how well can we model a system using a given measuring device? When is it possible to see *past* a measuring device into a system, and how can we optimize our choice of measuring device in practice?

Acknowledgments We thank Cina Aghamohammadi, Alec Boyd, David Darmon, Chris Ellison, Ryan James, John Mahoney, and Paul Riechers for helpful comments and the Santa Fe Institute for its hospitality during visits. JPC is an SFI External Faculty member. This material is based upon work supported by, or in part by, the U. S. Army Research Laboratory and the U. S. Army Research Office under contract numbers W911NF-12-1-0234, W911NF-13-1-0390, and W911NF-13-1-0340. NB was partially supported by NSF VIGRE Grant DMS0636297.

Appendix: Equivalence of Two ϵ-Transducer Definitions

We show the equivalence of two different ϵ-transducer definitions, that presented in the main paper and an earlier version requiring additional assumptions. Since the ϵ-transducer is determined by its causal equivalence relation, we show that the respective equivalence relations are the same. The first is defined and discussed at length above and duplicated here for convenience.

Definition 1 The *causal equivalence relation* $\sim_\epsilon$ for channels is defined as follows:

$$\overleftarrow{(x,y)} \sim_\epsilon \overleftarrow{(x,y)}' \iff \\ \mathbb{P}\big(\overrightarrow{Y}\big|\overrightarrow{X},\ \overleftarrow{(X,Y)} = \overleftarrow{(x,y)}\big) \\ = \mathbb{P}\big(\overrightarrow{Y}\big|\overrightarrow{X}, \overleftarrow{(X,Y)} = \overleftarrow{(x,y)}'\big)$$

The second definition is an implicit equivalence relation consisting of an explicit equivalence relation, along with an additional unifilarity constraint that, of course, is quite strong [48,49]. Here, we make both requirements explicit.

Definition 2 The *single-symbol unifilar equivalence relation* $\sim_\epsilon^1$ for channels is defined as follows:

$$\overleftarrow{(x,y)} \sim_\epsilon^1 \overleftarrow{(x,y)}' \iff \\ (i)\ \ \mathbb{P}\big(Y_0\big|X_0, \overleftarrow{(X,Y)}_0 = \overleftarrow{(x,y)}\big) \\ = \mathbb{P}\big(Y_0\big|X_0, \overleftarrow{(X,Y)}_0 = \overleftarrow{(x,y)}'\big) \\ \text{and:} \\ (ii)\ \ \mathbb{P}\big(Y_1\big|X_1, \overleftarrow{(X,Y)}_0 = \overleftarrow{(x,y)}, (X,Y)_0 = (a,b)\big) \\ = \mathbb{P}\big(Y_1\big|X_1, \overleftarrow{(X,Y)}_0 = \overleftarrow{(x,y)}', (X,Y)_0 = (a,b)\big),$$

for all $a \in \boldsymbol{\mathcal{X}}$ and $b \in \boldsymbol{\mathcal{Y}}$ such that:

$$\mathbb{P}\big((X,Y)_0 = (a,b)|\overleftarrow{(x,y)}\big) > 0$$

and:

$$\mathbb{P}\left((X,Y)_0 = (a,b)|\overleftarrow{(x,y)}'\right) > 0.$$

The second requirement (ii) in the above definition requires that appending any joint symbol to two single-symbol-equivalent pasts will also result in a pair of pasts that are single-symbol-equivalent. This is unifiliarity. The second part of the second requirement ensures that we are only considering *possible* joint symbols (a,b)—symbols that can follow $\overleftarrow{(x,y)}$ or $\overleftarrow{(x,y)}'$ with some nonzero probability.

Proposition 7 *The single-symbol unifilar equivalence relation is identical to the causal equivalence relation.*

Proof Let $\overleftarrow{(x,y)}$ and $\overleftarrow{(x,y)}'$ be two pasts, equivalent under $\sim_\epsilon^1$. This provides our base case for induction:

$$\mathbb{P}\big(Y_0\big|X_0, \overleftarrow{(x,y)}\big) = \mathbb{P}\big(Y_0\big|X_0, \overleftarrow{(x,y)}'\big). \tag{13}$$

Now, let's assume that $\overleftarrow{(x,y)}$ and $\overleftarrow{(x,y)}'$ are equivalent for length-$L-1$ future morphs:

$$\mathbb{P}\big(Y_{0:L}\big|X_{0:L}, \overleftarrow{(x,y)}\big) = \mathbb{P}\big(Y_{0:L}\big|X_{0:L}, \overleftarrow{(x,y)}'\big). \tag{14}$$

We need to show that $\overleftarrow{(x,y)}$ and $\overleftarrow{(x,y)}'$ are equivalent for length-L future morphs by using the unifilarity constraint. Unifilarity requires that appending a joint symbol (a,b) to both $\overleftarrow{(x,y)}$ and $\overleftarrow{(x,y)}'$ results in two new pasts also equivalent to each other for length-$L-1$ future morphs:

$$\mathbb{P}\big(Y_{1:L+1}\big|X_{1:L+1}, \overleftarrow{(x,y)}(a,b)\big) = \mathbb{P}\big(Y_{1:L+1}\big|X_{1:L+1}, \overleftarrow{(x,y)}'(a,b)\big). \tag{15}$$

Since this must be true for *any* joint symbol, we replace (a,b) with $(X,Y)_0$ in Eq. (15), giving:

$$\begin{aligned}
&\mathbb{P}\big(Y_{1:L+1}\big|X_{1:L+1}, \overleftarrow{(x,y)}, (X,Y)_0\big) = \mathbb{P}\big(Y_{1:L+1}\big|X_{1:L+1}, \overleftarrow{(x,y)}', (X,Y)_0\big)\\
&\quad\Longleftrightarrow\\
&\mathbb{P}\big(Y_{1:L+1}\big|X_{1:L+1}, \overleftarrow{(x,y)}, X_0, Y_0\big) = \mathbb{P}\big(Y_{1:L+1}\big|X_{1:L+1}, \overleftarrow{(x,y)}', X_0, Y_0\big).
\end{aligned} \tag{16}$$

To arrive at our result, we need to multiply the left side of Eq. (16) by $\mathbb{P}\big(Y_0\big|X_{1:L+1}, \overleftarrow{(x,y)}, X_0\big)$ and the right side by $\mathbb{P}\big(Y_0\big|X_{1:L+1}, \overleftarrow{(x,y)}', X_0\big)$, which we can do when these quantities are equal. Since our channel is *causal*, $X_{1:L+1}$ has no effect on Y_0 when we condition on the infinite joint past and present input symbol. The two quantities, $\mathbb{P}\big(Y_0\big|X_{1:L+1}, \overleftarrow{(x,y)}, X_0\big)$ and $\mathbb{P}\big(Y_0\big|X_{1:L+1}, \overleftarrow{(x,y)}', X_0\big)$, therefore reduce to $\mathbb{P}\big(Y_0\big|\overleftarrow{(x,y)}, X_0\big)$ and $\mathbb{P}\big(Y_0\big|\overleftarrow{(x,y)}', X_0\big)$, respectively. But these are equal by the single-symbol unifilar equivalence relation—the base for induction. Multiplying each side of Eq. (16) by these two terms (in their original form) gives:

$$\begin{aligned}
&\mathbb{P}\big(Y_{1:L+1}\big|X_{1:L+1}, \overleftarrow{(x,y)}, X_0, Y_0\big) \times \mathbb{P}\big(Y_0\big|X_{1:L+1}, \overleftarrow{(x,y)}, X_0\big)\\
&\quad = \mathbb{P}\big(Y_{1:L+1}\big|X_{1:L+1}, \overleftarrow{(x,y)}', X_0, Y_0\big) \times \mathbb{P}\big(Y_0\big|X_{1:L+1}, \overleftarrow{(x,y)}', X_0\big)\\
&\quad\Longleftrightarrow\\
&\mathbb{P}\big(Y_{1:L+1}, Y_0\big|X_{1:L+1}, \overleftarrow{(x,y)}, X_0\big) = \mathbb{P}\big(Y_{1:L+1}, Y_0\big|X_{1:L+1}, \overleftarrow{(x,y)}', X_0\big)\\
&\quad\Longleftrightarrow\\
&\mathbb{P}\big(Y_{0:L+1}\big|X_{0:L+1}, \overleftarrow{(x,y)}\big) = \mathbb{P}\big(Y_{0:L+1}\big|X_{0:L+1}, \overleftarrow{(x,y)}'\big).
\end{aligned}$$

The two pasts are therefore equivalent for length-L future morphs. By induction, the two pasts are equivalent for arbitrarily long future morphs. □

References

1. Crutchfield, J.P.: Between order and chaos. Nat. Phys. **8**(January), 17–24 (2012)
2. Packard, N.H., Crutchfield, J.P., Farmer, J.D., Shaw, R.S.: Geometry from a time series. Phys. Rev. Lett. **45**, 712 (1980)
3. Mandal, D., Jarzynski, C.: Work and information processing in a solvable model of Maxwell's Demon. Proc. Natl. Acad. Sci. USA **109**(29), 11641–11645 (2012)
4. Boyd, A.B., Crutchfield, J.P.: Demon dynamics: deterministic chaos, the Szilard map, and the intelligence of thermodynamic systems. Santa Fe Institute Working Paper 15–06-019. arXiv:1506.04327 [cond-mat.stat-mech]
5. Boyd, A.B., Mandal, D., Crutchfield, J.P.: Identifying functional thermodynamics in autonomous Maxwellian ratchets. Santa Fe Institute Working Paper 15-07-025. arXiv:1507.01537 [cond-mat.stat-mech]
6. Rieke, F., Warland, D., de Ruyter, R., van Steveninck, Bialek, W.: Spikes: Exploring the Neural Code. Bradford Book, New York (1999)

7. Cutsuridis, V., Hussain, A., Taylor, J.G.: Perception-Action Cycle. Springer, New York (2011)
8. Gordon, G., Kaplan, D., Lankow, D., Little, D., Sherwin, J., Suter, B., et al.: Toward an integrated approach to perception and action: conference report and future directions. Front. Syst. Neurosci. **5**, 20 (2011)
9. Padgett, J.F., Lee, D., Collier, N.: Economic production as chemistry. Ind. Corp. Chang. **12**(4), 843–877 (2003)
10. Crutchfield, J.P., Young, K.: Inferring statistical complexity. Phys Rev. Lett. **63**, 105–108 (1989)
11. Gray, R.M.: Probability, Random Processes, and Ergodic Theory, 2nd edn. Springer, New York (2009)
12. Cover, T.M., Thomas, J.A.: Elements of Information Theory, 2nd edn. Wiley, New York (2006)
13. Yeung, R.W.: Information Theory and Network Coding. Springer, New York (2008)
14. Ellison, C.J., Mahoney, J.R., Crutchfield, J.P.: Prediction, retrodiction, and the amount of information stored in the present. J. Stat. Phys. **136**(6), 1005–1034 (2009)
15. Crutchfield, J.P., Ellison, C.J., Mahoney, J.R., James, R.G.: Synchronization and control in intrinsic and designed computation: an information-theoretic analysis of competing models of stochastic computation. CHAOS **20**(3), 037105 (2010)
16. Gray, R.M., Davisson, L.D.: The ergodic decomposition of stationary discrete random processses. IEEE Trans. Inf. Theory **20**(5), 625–636 (1974)
17. Billingsley, P.: Statistical methods in Markov chains. Ann. Math. Stat. **32**, 12 (1961)
18. Crutchfield, J.P., Shalizi, C.R.: Thermodynamic depth of causal states: objective complexity via minimal representations. Phys. Rev. E **59**(1), 275–283 (1999)
19. Shalizi, C.R., Crutchfield, J.P.: Computational mechanics: pattern and prediction, structure and simplicity. J. Stat. Phys. **104**, 817–879 (2001)
20. Crutchfield, J.P., Ellison, C.J., Mahoney, J.R.: Time's barbed arrow: irreversibility, crypticity, and stored information. Phys. Rev. Lett. **103**(9), 094101 (2009)
21. N. Travers and J. P. Crutchfield. Equivalence of history and generator ϵ-machines. Santa Fe Institute Working Paper 11–11-051. arXiv:1111.4500 [math.PR]
22. Crutchfield, J.P.: The calculi of emergence: computation, dynamics, and induction. Physica D **75**, 11–54 (1994)
23. Paz, A.: Introduction to Probabilistic Automata. Academic Press, New York (1971)
24. Hopcroft, J.E., Ullman, J.D.: Introduction to Automata Theory, Languages, and Computation. Addison-Wesley, Reading (1979)
25. Lind, D., Marcus, B.: An Introduction to Symbolic Dynamics and Coding. Cambridge University Press, New York (1995)
26. Shannon, C.E.: A mathematical theory of communication. Bell Syst. Tech. J. **27**(379–423), 623–656 (1948)
27. Gray, R.M.: Entropy and Information Theory. Springer, New York (1990)
28. Oppenheim, A.V., Schafer, R.W.: Discrete-Time Signal Processing. Prentice-Hall Signal Processing Series. Prentice Hall, Upper Saddle River (2010)
29. Mandelbrot, B.: Fractals: Form. Chance and Dimension. W. H. Freeman and Company, New York (1977)
30. Rabiner, L.R.: A tutorial on hidden Markov models and selected applications. IEEE Proc. **77**, 257 (1989)
31. Elliot, R.J., Aggoun, L., Moore, J.B.: Hidden Markov Models: Estimation and Control. Applications of Mathematics, vol. 29. Springer, New York (1995)
32. James, R.G., Mahoney, J.R., Ellison, C.J., Crutchfield, J.P.: Many roads to synchrony: natural time scales and their algorithms. Phys. Rev. E **89**, 042135 (2014)
33. Crutchfield, J.P., Hanson, J.E.: Turbulent pattern bases for cellular automata. Physica D **69**, 279–301 (1993)
34. Li, C.-B., Yang, H., Komatsuzaki, T.: Multiscale complex network of protein conformational fluctuations in single-molecule time series. Proc. Natl. Acad. Sci. USA **105**, 536–541 (2008)
35. Varn, D.P., Canright, G.S., Crutchfield, J.P.: ϵ-Machine spectral reconstruction theory: a direct method for inferring planar disorder and structure from X-ray diffraction studies. Acta. Cryst. Sec. A **69**(2), 197–206 (2013)
36. Strelioff, C.C., Crutchfield, J.P.: Bayesian structural inference for hidden processes. Phys. Rev. E **89**, 042119 (2014)
37. Crutchfield, J.P., McNamara, B.S.: Equations of motion from a data series. Complex Syst. **1**, 417–452 (1987)
38. Johnson, B.D., Crutchfield, J.P., Ellison, C.J. McTague, C.S.: Enumerating finitary processes. Santa Fe Institute Working Paper 10–11-027. arXiv:1011.0036 [cs.FL]
39. Shalizi, C.R., Shalizi, K.L., Crutchfield, J.P.: Pattern discovery in time series, Part I: Theory, algorithm, analysis, and convergence. Santa Fe Institute Working Paper 02–10-060. arXiv:cs.LG/0210025
40. Still, S., Crutchfield, J.P., Ellison, C.J.: Optimal causal inference: estimating stored information and approximating causal architecture. CHAOS **20**(3), 037111 (2010)

41. Crutchfield, J.P.: Reconstructing language hierarchies. In: Atmanspracher, H.A., Scheingraber, H. (eds.) Information Dynamics, pp. 45–60. Plenum, New York (1991)
42. Hanson, J.E., Crutchfield, J.P.: The attractor-basin portrait of a cellular automaton. J. Stat. Phys. **66**, 1415–1462 (1992)
43. Crutchfield, J.P.: Discovering coherent structures in nonlinear spatial systems. In: Brandt, A., Ramberg, S., Shlesinger, M. (eds.) Nonlinear Ocean Waves, pp. 190–216. World Scientific, Singapore (1992)
44. Hanson, J.E., Crutchfield, J.P.: Computational mechanics of cellular automata: An example. Physica D **103**, 169–189 (1997)
45. McTague, C.S., Crutchfield, J.P.: Automated pattern discovery—an algorithm for constructing optimally synchronizing multi-regular language filters. Theoe. Comp. Sci. **359**(1–3), 306–328 (2006)
46. Crutchfield, J.P., Görnerup, O.: Objects that make objects: the population dynamics of structural complexity. J. R. Soc. Interface **3**, 345–349 (2006)
47. Crutchfield, J.P., Whalen, S.: Structural drift: the population dynamics of sequential learning. PLoS Comput. Biol. **8**(6), e1002510 (2010)
48. J. P. Crutchfield. Optimal structural transformations-the ϵ-transducer. UC Berkeley Physics Research Report, 1994
49. Shalizi, C.R.: Causal architecture, complexity and self-organization in time series and cellular automata. PhD thesis, University of Wisconsin, Madison (2001)
50. Hopfield, J.J.: Neural networks and physical systems with emergent collective behavior. Proc. Natl. Acad. Sci. **79**, 2554 (1982)
51. Hertz, J., Krogh, A., Palmer, R.G.: An Introduction to the Theory of Neural Networks. Lecture Notes, vol. 1. Studies in the Sciences of Complexity. Addison-Wesley, Redwood City (1991)
52. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2nd edn. Spinger, New York (2011)
53. MacKay, D.J.C.: Information Theory. Inference and Learning Algorithms. Cambridge University Press, Cambridge (2003)
54. Bialek, W., Nemenman, I., Tishby, N.: Predictability, complexity, and learning. Neural Comput **13**, 2409–2463 (2001)
55. Shalizi, C.R., Crutchfield, J.P.: Information bottlenecks, causal states, and statistical relevance bases: How to represent relevant information in memoryless transduction. Adv. Comput. Syst. **5**(1), 91–95 (2002)
56. Albers, D., Sprott, J.C., Crutchfield, J.P.: Persistent chaos in high dimensions. Phys. Rev. E **74**(5), 057201 (2006)
57. Marzen, S., Crutchfield, J.P.: Informational and causal architecture of discrete-time renewal processes. Entropy **17**(7), 4891–4917 (2015)
58. Crutchfield, J.P.: Information and its metric. In: Lam, L., Morris, H.C. (eds.) Nonlinear Structures in Physical Systems—Pattern Formation. Chaos and Waves, pp. 119–130. Springer, New York (1990)
59. Shannon, C.E.: The lattice theory of information. IEEE Trans. Inf. Theory **1**, 105–107 (1953)
60. Li, H., Chong, E.K.P.: On a connection between information and group lattices. Entropy **13**, 683–798 (2011)
61. Morse, M., Hedlund, G.A.: Symbolic dynamics. Am. J. Math. **60**(4), 815–866 (1938)
62. Post, E.: Introduction to the general theory of elementary propositions. Am. J. Math. **43**, 163–185 (1921)
63. Gödel, K.: On Formally Undecidable Propositions of Principia Mathematica and Related Systems. Dover Publications, Mineola (1992)
64. Turing, A.: On computable numbers, with an application to the Entschiedungsproblem. Proc. Lond. Math. Soc. **42**, **43**:230–265, 544–546 (1937)
65. Church, A.: A note on the Entscheidungsproblem. J. Symb. Log. **1**, 40–41 (1936)
66. Moore, E.F.: Gedanken-experiments on sequential machines. In: Shannon, C., McCarthy, J. (eds.) Automata Studies. number 34 in Annals of Mathematical Studies, pp. 129–153. Princeton University Press, Princeton, New Jersey (1956)
67. Huffman, D.: The synthesis of sequential switching circuits. J. Frankl Inst. **257**(161–190), 275–303 (1954)
68. Huffman, D.: Information conservation and sequence transducers. In: Proceedings of the Symposium on Information Networks, pp. 291–307. Polytechnic Institute of Brooklyn, Brooklyn (1954)
69. Huffman, D.: Canonical forms for information-lossless finite-state logical machines. IRE Trans. Circuit Theory **6**, 41–59 (1959)
70. Huffman, D.: Notes on information-lossless finite-state automata. Il Nuovo Cimento **13**(2 Supplement), 397–405 (1959)
71. Brookshear, J.G.: Theory of Computation: Formal Languages, Automata, and Complexity. Benjamin/Cummings, Redwood City (1989)
72. Ash, R.B.: Information Theory. Wiley, New York (1965)

73. Blackwell, D., Breiman, L., Thomasian, A.J.: Proof of Shannon's transmission theorem for finite-state indecomposable channels. Ann. Math. Stat. **29**(4), 1209–1220 (1958)
74. Blackwell, D.: Exponential error bounds for finite state channels. In: Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, vol. 1, pp. 57–63. University of California Press, (1961)
75. Hamming, R.W.: Digital Filterns, 3rd edn. Dover Publications, New York (1997)
76. Bateson, G.: Mind and Nature: A Necessary Unity. E. P. Dutton, New York (1979)
77. R. Shaw. Strange attractors, chaotic behavior, and information flow. Z. Naturforsh., 36a:80, 1981
78. Ahlswede, R., Körner, J.: Appendix: on common information and related characteristics of correlated information sources. In: Ahlswede R., Baumer, Cai N., Aydinian H., Blinovsky V., Deppe C., Mashurian H. (eds.) General Theory of Information Transfer and Combinatorics. Lecture Notes in Computer Science, vol. 4123, pp. 664–677. Springer, Berlin (2006)
79. Williams, P.L., Beer, R.D.: Generalized measures of information transfer. arXiv:1102.1507
80. Barnett, L., Barrett, A.B., Seth, A.K.: Granger causality and transfer entropy are equivalent for Gaussian variables. Phys. Rev. Lett. **103**, 238701 (2009)
81. Schreiber, T.: Measuring information transfer. Phys. Rev. Lett. **85**, 461–464 (2000)
82. Marko, H.: The bidirectional communication theory—a generalization of information theory. IEEE Trans. Commun. **21**, 1345–1351 (1973)
83. Massey, J.L.: Causality, feedback and directed information. In: Proceedings of the 1990 International Symposium on Information Theory and Its Applications, pp. 1–6. Waikiki, Hawaii, 27–30 Nov 1990
84. Sun, J., Bollt, E.M.: Causation entropy identifies indirect influences, dominance of neighbors and anticipatory couplings. Physica D **267**, 49–57 (2014)
85. Barnett, N., Crutchfield, J.P.: Computational mechanics of input-output processes: Shannon information measures and decompositions. in preparation (2014)
86. Fedorov, V.V.: Theory of Optimal Experiments. Probability and Mathematical Statistics. Academic Press, New York (1972)
87. Atkinson, A., Bogacka, B., Zhigljavsky, A.A. (eds.): Optimum Design 2000. Nonconvex Optimization and Its Applications. Springer, New York (2001)
88. Still, S.: Information-theoretic approach to interactive learning. EuroPhys. Lett. **85**, 28005 (2009)
89. Astrom, K.J., Murray, R.M.: Feedback Systems: An Introduction for Scientists and Engineers. Princeton University Press, Princeton (2008)
90. Crutchfield, J.P., Riechers, P., Ellison, C.J.: Exact complexity: spectral decomposition of intrinsic computation. Santa Fe Institute Working Paper 13–09-028. arXiv:1309.3792 [cond- mat.stat-mech]