Some first principles of complex systems theory

Axel G. Rossberg

Yokohama National University, Japan
axel@rossberg.net

A general goal of complex systems theory is to find simple, efficient, and accurate descriptions of empirical complex systems. For a general theory, the notion of *accuracy* or, for that matter, *approximation* needs to be generalized. A network topology will require different kinds of approximations than a time series. This requires a distinction between characteristic properties which should be contained in the description from others which are unimportant. Here, a rigorous formalism for objectively making such a distinction is developed by applying concepts from computer science and statistics to an idealized, generic, computer-controlled experiment.

1   Introduction

A complex system can become an economic problem. Understanding its internal machinery, describing it, and predicting its future behaviour can be expensive. The problem of finding simple, accurate, and efficient descriptions is a central aspect of the work on complex systems. Perhaps it is *the* unifying aspect of complex-systems science.

Interestingly, this practical problem is closely related to the philosophical problem of emergence [1, 2]. Stated in its weakest form, this is the question why, if the basic laws of physics are so simple, the world around us appears to have such a rich structure. A partial answer that easily comes to mind is this: If we would try to apply the basic laws every time we interpret the world around us, it would just take too much time. Instead, we are using other descriptions that are more efficiently. But each applies only to a particular part of the world; so we need many of them. In the language of computer science [3], we are trading computation time for description length. Apparently, this is a good deal. The structure of the world as we see it is a result of solving just the economic problem mentioned above. We are reducing the cost of describing the complex system "world".

This is only a partial answer to the problem of emergence. Many questions remain unanswered, such as, "Why are there distinct parts for which efficient descriptions exist?" or "Can efficient descriptions be found systematically, and, if

yes, how?". But it is this partial answer that will be of interest here, for it is in itself incomplete.

Efficient, simplified descriptions are rarely perfectly precise, and somehow a decision has to be made which information about the thing described the description should reproduce, and which may be ignored. The conventional strategy to proceed when arriving at this part of the problem (e.g. [4, 5, 6]) is to presupposed that the information regarding the thing described is incomplete anyway, and only the available information must be reproduced. This blurring of the picture comes under many different names: finite samples of noisy data, coarse graining, partitioning of the state space, *etc.* As a result, the choice of the simplified description becomes essentially a function of the mode of observation. But does this correspond to the facts? The history of science knows many examples of simplified descriptions (and related concepts) that have been introduced long before the things described could be observed. Obvious examples are descriptions in terms of quasi-particles such as "holes" and "phonons" used in solid state physics. On the other hand, descriptions that are much coarser than any reasonable limit of observation are also frequently used. One might just think of a description of traffic flow in terms of atomic "cars".
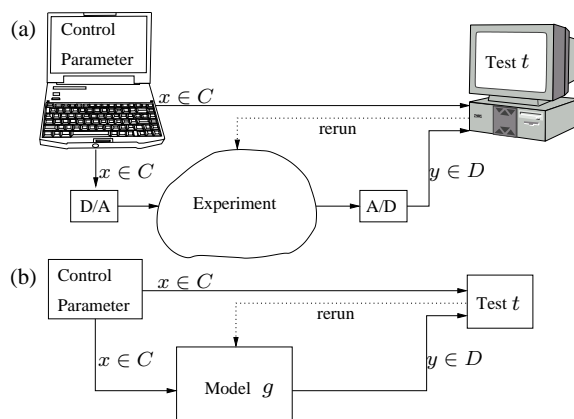
Shalizi and Moore [7] suggested a solution of this problem based on *causal states* [8]. Here, a different argument for reducing the information to be reproduced by a description is explored. Information regarding the thing described is dropped not because it is unavailable, but for the sake of an efficient and simple description. Central to this argument is the distinction between two kinds of descriptions: *models*, that produce data somehow similar to present or future real data, and *characterizations* that summarize some aspects of data.

Predictions about complex systems generally require both: a model that is used for the prediction, and a characterization that specifies what aspects of the real data the model is supposed to reproduce. By the condition that model and characterization are *both* simple and efficient, particular choices for the information to be retained by the descriptions are singled out. This part of the information is "relevant" for a simple reason: it can be predicted within given cost constraints.

In the remainder of this work, it is shown that this approach can be taken beyond hand-waving. Formal definitions of basic notions are introduced. Desiderata for economic descriptions are summarized under the notion of *basic model-specifying characterizations* (b.m.s.c.), and it is shown that nontrivial b.m.s.c. exist. They are by far not unique. The accuracy and detail of preferred descriptions depends on the available resources, and the formalism is taking this into account. Results are illustrated by a minimal example.

2   The formalism

For the formal analysis, both models and characterizations are represented by computer programs. The complex system to be described is represented by a computer-controlled experiment. Fig. 1 illustrates the interaction between experimenter (the "Control Parameter" terminal), experiment, model, and characterization. A characterization of data is given by a statement saying the data passes

Figure 1: (a) Generic setup of a computer-controlled experiment. (b) Data flow in a test of a computational model.

a certain *test*; a statistical test in general.

Throughout the theory, assume a **control parameter format** $C \subset \{0,1\}^n$ and a **data format** $D \subset \{0,1\}^m$ to be fixed, with $\{0,1\}^k$ denoting the set of all binary strings of length k and $n, m \in \mathbb{N}_0$. Given a control parameter value $x \in C$ and being run, the experiment (including the D/A and A/D conversion) produces an output value $y \in D$. Input and output data can be sets of numbers, images, time-series, *etc.* The only major limitation is that both C and D are finite sets. The A/D conversion of the experimental output naturally involves some loss of information. But below it is argued that the information passing through the A/D converter can be much richer than the information tested for and being reproduced in the model. The information loss at the A/D converter is not decisive for determining the "emergent" description.

In general, the complex system involved in the experiment is not deterministic. The experimental output y is a realization of a random variable Y with values in D. The experiment is assumed reproducible in the sense that repeated runs of the experiment (with identical x) yield a sequence $Y_1, Y_2, \ldots$ of statistically independent, identically distributed (i.i.d.) results.

Definition 1 *For a given (deterministic) machine model, a* **test** *t is a program that takes a control parameter $x \in C$ as input, runs, and then halts with output 0, 1, or* **e**. *When the output is not* **e**, *the test can request several data samples before halting ("rerun" in Fig. 1). Then, execution of the test is suspended until a sample $y \in D$ is written into a dedicated storage accessible by the test. The number of samples requested can depend on the sampled y but is finite for any sequence of successive samples.*

By the output **e** the tests t indicates that x is not within the range of validity $C[t] := \{x | x \in C$ and output of t with input x is not **e**$\}$ of the corresponding characterization. The outputs 1 or 0 indicate that the null hypothesis (see below) is accepted or rejected by the test, respectively.

Models are represented by *generators*.

Definition 2 *Given a machine model, a* **generator** *g is a program that takes a control parameter $x \in C$ as input, runs, outputs data $y \in D$ and halts. The program has access to a source of independent, evenly distributed random bits in an otherwise deterministic machine.*

Now, a cost functions is introduced which measures the cost involved in running models g and tests t, constructing and evaluating them, and performing experiments. We assume that this cost can be expressed in terms of the lengths $L(t), L(g) \in \mathbb{N}_0$ of the programs t and g, their average execution times $T(g), T(t) \in \mathbb{R}^{\geq 0}$, and the average number $N(t) \in \mathbb{R}^{\geq 0}$ of experimental runs required by t. To be specific, define $T(\cdot)$ as the maximum of the expectation value of the runtime over all $x \in C$ and all distributions of input data, $N(\cdot)$ analogously. It can be shown that $T(t)$ and $N(t)$ are always finite. As conventional, the number of tests or generators q with length $L(q) \leq n$ is assumed to be finite all $n \in \mathbb{N}_0$.

Definition 3 *A* **cost function** *K is a mapping $K : \mathbb{N}_0 \times \mathbb{R}^{\geq 0} \to \mathbb{R}^{\geq 0}$ or $K : \mathbb{N}_0 \times \mathbb{R}^{\geq 0} \times \mathbb{R}^{\geq 0} \to \mathbb{R}^{\geq 0}$ that increases strictly monotonically in all its arguments. The abbreviation $K(t)$ stands for $K[L(t), T(t), N(t)]$ if t is a test and $K(g)$ stands for $K[L(g), T(g)]$ if g is a generator.*

In practice, the cost of descriptions depends strongly on the circumstances. The theory should therefore be independent of the particular choice of the cost function. For this purpose, as is made clear by Theorem 3 below, the following definition is convenient.

Definition 4 *Let p_1 and p_2 be two tests or two generators. Then the relations $\preceq$ (**always cheaper or equal**) and $\prec$ (**always cheaper**) are defined by*

$$p_1 \preceq p_2 \stackrel{def}{\Leftrightarrow} L(p_1) \leq L(p_2) \text{ and } T(p_1) \leq T(p_2) \text{ and } N(p_1) \leq N(p_2) \qquad (1)$$

(for generators without the last condition) and

$$p_1 \prec p_2 \stackrel{def}{\Leftrightarrow} p_1 \preceq p_2 \text{ and not } p_2 \preceq p_1. \qquad (2)$$

A test or generator p is said to be $\prec$**-minimal** in a set P of tests or generators if $p \in P$ and there is no $p' \in P$ such that $p' \prec p$.

Lemma 1 *Relation $\preceq$ is transitive and reflexive, relation $\prec$ is transitive and antireflexive.*

(Since $\preceq$ is not antisymmetric, it is not a partial order.) The proof is standard.

Lemma 2 *For any two tests or generators p_1, p_2, and any cost function K, $p_1 \prec p_2$ implies $K(p_1) < K(p_2)$.*

PROOF   Assume that p_1 and p_2 are generators. Then $L(p_1) \leq L(p_2)$ and $T(p_1) \leq T(p_2)$ and either $L(p_1) < L(p_2)$ or $T(p_1) < T(p_2)$, since if both were equal the last part of condition (2) would be violated. Thus, using the strict monotony of K, one has either $K[L(p_1), T(p_1)] < K[L(p_2), T(p_1)] \leq K[L(p_2), T(p_2)]$ or $K[L(p_1), T(p_1)] \leq K[L(p_2), T(p_1)] < K[L(p_2), T(p_2)]$. Both imply $K(p_1) < K(p_2)$. For tests the proof is analogous. ∎

Theorem 3 *Let P be a set of tests or generators. $p \in P$ is $\prec$-minimal in P if and only if there is a cost function K that attains its minimum over P at p.*

PROOF   The "if" part: If some K would attain its minimum over P at p but p was not $\prec$-minimal, there would be a $p' \in P$ such that $p' \prec p$ and, by Lemma 2, $K(p') < K(p)$. But this contradicts the premise. So p is $\prec$-minimal.

The "only if" part: Assume p is $\prec$-minimal in a set of generators P. We show that there is a cost function that attains its minimum over P at p by explicit construction. Let $K(l, t) := \kappa(l, L(p)) + \kappa(t, T(p))$ with $\kappa(z, z_0) = z$ for $z \leq z_0$ and $\kappa(z, z_0) = L(p) + T(p) + z$ for $z > z_0$. Obviously K satisfies strict monotony. And any $p' \in P$ that does not have $L(p') = L(p)$ and $T(p') = T(p)$ [and hence $K(p') = K(p)$] must have either a larger L or a larger T than p, otherwise p would not be $\prec$-minimal. But then $K(p') \geq L(p) + T(p) = K(p)$. So $K(p)$ is the minimum of K over P. For tests the proof is analogous. ∎

Lemma 4 *Every nonempty set P of tests or generators contains an element p which is $\prec$-minimal in P.*

PROOF   Assume that P has no $\prec$-minimal element. Then for every element $p \in P$ there is a $p' \in P$ such that $p' \prec p$. Thus an infinite sequence of successively always-cheaper ($\prec$) elements of P can be constructed. Because $\prec$ is transitive and antireflexive, such a sequence contains each element at most once. Let q be the first element of such a sequence. Since, by definition, $p \prec q$ implies $L(p) \leq L(q)$, and there is only a finite number of programs p with $L(p) \leq L(q)$, the number of successors of q cannot be infinite. So the premise that P has no $\prec$-minimal element is wrong for any nonempty P. ∎

The $\prec$-minimal element is generally not unique. Different $\prec$-minima minimize cost functions that give different weight to the resources length, time, and, experimental runs used. On the other hand, it turns out that in practice the machine-dependence of relation $\prec$ for implementations of algorithms on different processor models is weak. Therefore, instead of cost functions, relation $\prec$ is used below.

A central element of statistical test theory [9] is the *power function*. It is defined as the probability that the test rejects data of a given (usually parameterized) distribution. The goal of statistical test theory is to find tests whose power function is below a given significance level α if the null-hypothesis is satisfied, and as large as possible otherwise.

Denote by the **test function** $t_x(\{y_i\})$ the output of the test t at control parameter $x \in C[t]$ when applied to the sequence of experimental results $\{y_i\} \in D^\infty$ (for formal simplicity, the sequences $\{y_i\}$ are assumed infinite, even though tests use only finite subsequences).

Definition 5 *For any test t, the **power** of the test function t_x, when applied to the random sequence $\{Y_i\}$ with values in D^∞, is the probability of rejecting $\{Y_i\}$, i.e.,*

$$\mathrm{pow}(t_x, \{Y_i\}) := \Pr[t_x(\{Y_i\}) = 0] \quad (x \in C[t]). \tag{3}$$

Unlike in conventional test theory, there is no independent null hypothesis H_0 here that states the distribution or the class of distributions of $\{Y_i\}$ that is tested for. Instead, given a test function t_x, the null hypothesis, i.e., the class of distributions, is *defined* by the condition

$$\text{pow}(t_x, \{Y_i\}) \leq \alpha, \tag{4}$$

where $0 < \alpha < 1$ is a fixed[1] significance level.

Now, the concepts from statistics and computer science introduced above are combined. Denote by g_x the sequence $\{Y_i\}$ of random outputs of generator g at control parameter x.

Definition 6 *A generator g is an* **optimal generator** *relative to a test t and a power threshold $1 > \gamma > \alpha$ (notation: $\text{opt}_t^\gamma g$) if*

  1. $\text{pow}(t_x, g_x) \leq \alpha$ for all $x \in C[t]$ and

  2. for every generator $g' \prec g$ there is a $x \in C[t]$ such that $\text{pow}(t_x, g_x') > \gamma$.

This implies that g is $\prec$-minimal in $\{g' | \text{pow}(t_x, g_x') \leq \alpha \text{ for all } x \in C[t]\}$. Hence g is, for some cost function, the minimal (-cost) model for the property that t is testing for. The second condition can be satisfied only for particular choices of t. It requires a minimal power γ from t to distinguish the models that it characterizes from those is does not. Constructing tests that maximize γ leads to results similar to the *locally most powerful tests* of statistical test theory [9].

For an i.i.d. random sequence $\{Y_i\}$ denote by $p[\{Y_i\}]$ the distribution function of its elements, i.e., $p[\{Y_i\}](y) := \Pr[Y_1 = y]$ for $y \in D$.

Definition 7 *Call a generator g an* **optimal implementation** *with respect to a set $\tilde{C} \subset C$ if it is $\prec$-minimal in $\{g' | p[g_x'] \equiv p[g_x] \text{ for all } x \in \tilde{C}\}$ (the set of generators that do exactly the same).*

Theorem 5 *For every $\tilde{C} \subset C$, every optimal implementation g with respect to $\tilde{C}$, and every $1 > \gamma > \alpha$ there is, a test t such that $\text{opt}_t^\gamma g$ and $C[t] = \tilde{C}$.*

PROOF   An explicit construction of t is outlined: $x \in \tilde{C}$ can be tested for by keeping a list of $\tilde{C}$ in t. Since there is only a finite number of $g' \preceq g$, the test must distinguish $p[g_x]$ from a finite number of different distributions $p[g_x']$ for all $x \in \tilde{C}$, with power γ. This can be achieved by comparing a sufficiently accurate representation of $p[g_x]$, stored in t for all $x \in \tilde{C}$, with a histogram obtained from sufficiently many samples of g_x'. ∎

Definition 8 *Call a pair (t, g) a* **basic model-specifying characterization** *(b.m.s.c.) if t is $\prec$-minimal in $\{t' | \text{opt}_{t'}^\gamma g \text{ and } C[t] \subset C[t']\}$ for some $1 > \gamma > \alpha$.*

---

[1]From t_x, tests for the same H_0 at other significance levels can be constructed.

That is, for some cost function the test t gives the minimal characterization required to specify g (given power threshold γ and range of validity $C[t]$). Sometimes there are other generators which are similar to g but cheaper. Then t must be very specific to characterize the particularities of g. In other cases, the output of g has an essentially new, "striking" property which cannot be obtained with cheaper generators. If the property is really "striking", a rather cheap and generic test t is sufficient to detect it. *Thus t can ignore all other information contained in the output of g.* Such an approximate characterization is most likely to apply also to the data of an actual experiment. Then the b.m.s.c. (t, g) provides a specific but economic description. After verifying the b.m.s.c. for some control parameters $x \in C[t]$, approximate predictions of experimental results for other parameters can be obtained from g by the usual (though philosophically opaque) method of induction.

A trivial b.m.s.c. is given by a test t that always outputs 1 and some generator $g \prec$-minimal among all generators. But the following makes clear that the world of b.m.s.c. is much richer.

Theorem 6 *There is, for every $\tilde{C} \subset C$ and every optimal implementation g with respect to $\tilde{C}$, a test t such that (t, g) is a b.m.s.c. and $\tilde{C} \subset C[t]$.*

PROOF   Fix some $1 > \gamma > \alpha$. By Theorem 5, the set $S := \{t' | \mathrm{opt}_{t'}^{\gamma} g$ and $\tilde{C} \subset C[t']\}$ is nonempty. Theorem 6 is satisfied by any t which is $\prec$-minimal in S. By Lemma 4, such an element exists. ∎

3   A simple example

As a minimal, analytically traceable example, consider an experiment without control parameters $C = \emptyset$ in which only a single bit is measured, $D = \{0, 1\}$. The probability p for the cases $y = 0$ to occurs is exactly $p = 0.52$, and the "complexity" of the systems consists just in this nontrivial value. With $\alpha = 0.1$, the following pair (t, g) is a b.m.s.c.: A generator g [with $L(g) = 52$ byte and $T(g) = 56\,\upsilon$ on the MMIX model processor [10]; the unit of time reads "oops"] that outputs $y = 0$ and $y = 1$ with exactly equal probability $p = 1/2$, and a test t ($L(t) = 104$ byte and $T(t) = 255\,\upsilon$) that verifies if among $N = 5$ samples both $y = 0$ and $y = 1$ occur at least once. This test is the cheapest test that accepts the model g ($\mathrm{pow}(t, \{g\}) = 1/16 \leq \alpha$) and rejects all cheaper models, namely generators g' that always output the same value [one finds $L(g') = 28$ byte, $T(g') = 38\,\upsilon$, $\mathrm{pow}(t, \{g'\}) = 1 > \alpha$]. But t also characterizes all experiments for which $\mathrm{pow}(t, \{Y_i\}) = p^N + (1 - p)^N \leq \alpha$, such as our case $p = 0.52$, where $\mathrm{pow}(t, \{Y_i\}) \approx 0.064$.

There are other b.m.s.c. for the experiment. For example, a generator g^* that computes a 8-bit random integer in the range $0, ..., 2^8 - 1$, and uses it to output $y = 0$ with probability $p = 133 \times 2^{-8} = 0.5195$ and $y = 1$ otherwise [$L(g^*) = 76$ byte and $T(g^*) = 225\,\upsilon$]; and a test t^* that verifies if within 962 samples between 437 and 487 cases $y = 0$ occur [$L(t^*) = 112$ byte, $T(t^*) = 40430\,\upsilon$]. One finds $\mathrm{pow}(t^*, \{g^*\}) = 0.099834 \leq \alpha = 0.1$ and $\mathrm{pow}(t^*, \{Y_i\}) = 0.099832 \leq \alpha$ for the experimental data. The next cheapest generators, which have $p = 132 \times 2^{-8} =$

0.5156 or $p = 134 \times 2^{-8} = 0.5234$, and are faster because they require only 6-bit or 7-bit random numbers respectively, are rejected with a power larger than $\gamma = 0.108576 > \alpha$. A cheaper test could not reach this γ.

One might think of g, $g*$, and some exact g^{**}, as primitive forms of different levels of description for the same experiment.

Bibliography

[1] T. O'Connor and H. Y. Wong, "Emergent properties," in *The Stanford Encyclopedia of Philosophy* (E. N. Zalta, ed.), 2002.

[2] R. I. Damper, "Emergence and levels of abstraction," *International Journal of Systems Science*, vol. 31, no. 7, pp. 811–818, 2000. Editorial for the Special Issue on 'Emergent Properties of Complex Systems'.

[3] M. Li and P. Vitányi, *An introduction to Kolmogorov complexity and its applications.* New York: Springer, 2nd ed., 1997.

[4] E. Castellani, "Reductionism, emergence, and effective field theories," *Studies in History and Philosophy of Science Part B*, vol. 33, no. 2, pp. 251–267, 2002.

[5] J. P. Crutchfield, "The calculi of emergence: Computation, dynamics, and induction," *Physica D*, vol. 75, pp. 11–54, 1994.

[6] J. Rissanen, *Stochastic Complexity in Statistical Inquiry.* Singapore: World Scientific, 1989.

[7] C. R. Shalizi and C. Moore, "What is a macrostate? Subjective observations and objective dynamics." arXiv:cond-mat/0303625v1, 2000-2004.

[8] J. P. Crutchfield and K. Young, "Inferring statistical complexity," *Phys. Rev. Lett.*, no. 63, pp. 105–108, 1989.

[9] E. L. Lehmann, *Testing Statistical Hypotheses.* Heidelberg: Springer, 2nd ed., 1997.

[10] D. E. Knuth, *MMIXware: A RISC Computer for the Third Millennium.* No. 1750 in Lecture Notes in Computer Science, Heidelberg: Springer, 1999.