How Turing parasites expand the computational landscape of digital life

Luís F. Seoane[1,2] and Ricard Solé[3,4,5]

[1]*Departamento de Biología de Sistemas, Centro Nacional de Biotecnología (CSIC), C/Darwin 3, 28049 Madrid, Spain*
[2]*Grupo Interdisciplinar de Sistemas Complejos (GISC), Madrid, Spain*
[3]*ICREA-Complex Systems Lab, Universitat Pompeu Fabra (GRIB), Dr Aiguader 80, 08003 Barcelona, Spain*
[4]*Institut de Biologia Evolutiva, CSIC-UPF, Pg Maritim de la Barceloneta 37, 08003 Barcelona, Spain*
[5]*Santa Fe Institute, 1399 Hyde Park Road, Santa Fe, New Mexico 87501, USA*

(Received 8 February 2023; revised 10 August 2023; accepted 19 September 2023; published 23 October 2023)

Why are living systems complex? Why does the biosphere contain living beings with complexity features beyond those of the simplest replicators? What kind of evolutionary pressures result in more complex life forms? These are key questions that pervade the problem of how complexity arises in evolution. One particular way of tackling this is grounded in an algorithmic description of life: living organisms can be seen as systems that extract and process information from their surroundings to reduce uncertainty. Here we take this computational approach using a simple bit string model of coevolving agents and their parasites. While agents try to predict their worlds, parasites do the same with their hosts. The result of this process is that, to escape their parasites, the host agents expand their computational complexity despite the cost of maintaining it. This, in turn, is followed by increasingly complex parasitic counterparts. Such arms races display several qualitative phases, from monotonous to punctuated evolution or even ecological collapse. Our minimal model illustrates the relevance of parasites in providing an active mechanism for expanding living complexity beyond simple replicators, suggesting that parasitic agents are likely to be a major evolutionary driver for biological complexity.

DOI: 10.1103/PhysRevE.108.044407

I. INTRODUCTION

With the emergence of life on Earth, information processing took on an unprecedented relevance [1–5]. Swiftly, mechanisms for error correction [5], memory (hence path dependency and contingency) [6,7], and the capacity to predict the environment [8] rose up. These processes were tightly linked to the arrival of autonomous agents [5,9], setting up the stage for evolution of complexity through natural selection [9–11]—an algorithmic process itself [11,12] deeply related to information theory [13–15].

A crucial question here is whether this process can generate open-ended novelty [16–21] and what kind of universal patterns might be involved [22]. In this context, a key question is what drives the rise of organismal complexity beyond simple replicators. One view sustains that selection has no intrinsic bias toward complex life forms [23]. According to this view, bacteria constitute the peak performance dominating the evolutionary landscape. Organisms much simpler would not be viable, thus prevailing deviations from the peak fitness would, in average, look like an increase in complexity even if no bias favoring greater complexity exists. A naive rendering of this hypothesis suggests a single-peaked distribution of abundance across the spectrum of life's complexity. Is that the case?

Along with cognition [24], parasites have been proposed as causal agents that boost biological complexity, grounded in a plethora of studies, particularly regarding viruses [25–27]. In general, parasites of very different nature are known to have a major impact on their host's survival [28]. This has inspired theoretical and computational efforts to explain how parasites can affect the genetic structure of host populations. *In silico* models [29–31] have helped explore a series of essential questions concerning parasitic dynamics. These include the emergence of sex as a mechanism of resistance against parasites [32], improving the efficiency of evolutionary algorithms [33], or parasites as promoters of mutualism [34]. *In silico* evolution models show how parasites can promote evolvability and the complexity of their hosts, an effect notoriously driven by the parasite's memory [35]. These models share a common strategy of using digital genomes and simplified case studies to gain the highest insight. This logical description is an essential part of biology, and a computational picture of interaction can capture much more than we would initially expect [36,37].

In this paper we use such a formal description, with parasites and their hosts treated as abstract machines [38]. Our model is inspired by the Turing machine model of computation [39] where an abstract computing device is used as a way to formalize computation in terms of a simple automaton that reads a binary tape. Despite its simplicity, this kind of formal approach has been used to explore the limits of information processing in both physical and biological systems [40]. Unlike that original framework, our "Turing parasites" deal with stochastic environments that they need to predict. Under the right conditions (which require sufficiently successful predictive power) they can replicate themselves. Moreover, they can contract or expand their computational capabilities if evolutionary change is allowed. As shown below, parasites can act as complexity enhancers in an open-ended manner. Our formulation of environments, hosts, and parasites as strings of bits with minimal rules brings our model very close to

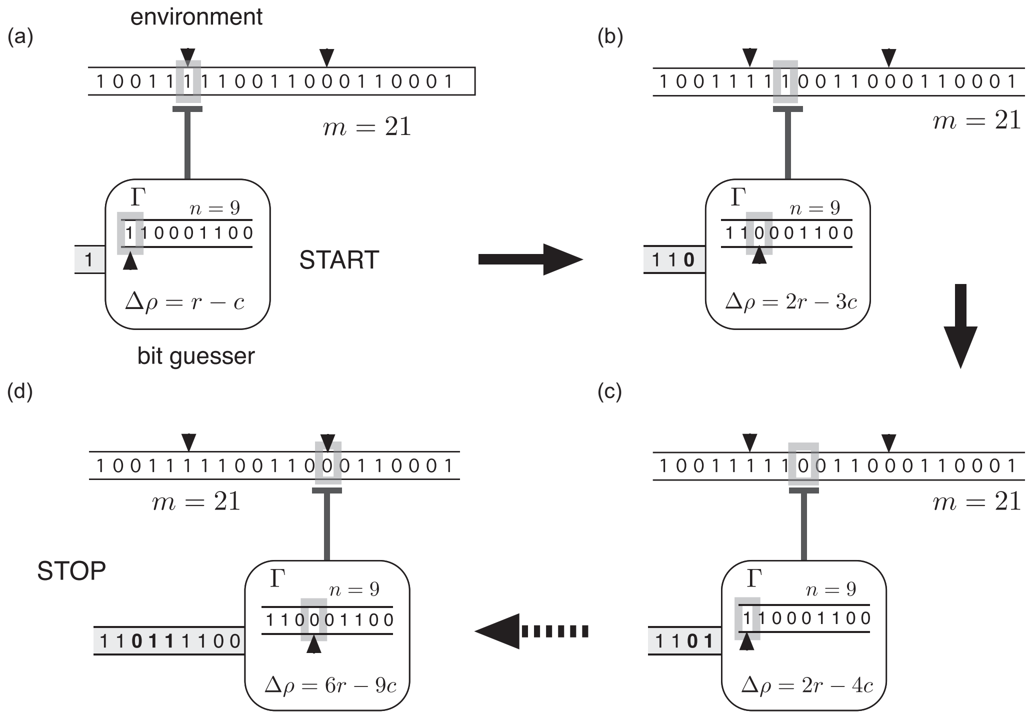

FIG. 1. A bit-guesser "machine" computing its environment. Bit guessers use a bit-string (Γ) to attempt to predict their environment. Here Γ approximates the most likely pattern to be found in the environment E—e.g., the first bit proposed (a) is just the most common bit in E. For each guess attempt, a cost c is subtracted from guesser's reward ρ. For each correct guess, a reward r is added. (b) After a guess, the guesser moves forward along E. If the guess is correct, then the guesser keeps sampling Γ in linear order. If the guess is wrong [(b, c), bits highlighted in bold], then the agent resets to the beginning of Γ for the next guess attempt. (d) The process goes on until as many bits have been visited as the size of the guesser's internal pattern Γ (in this case, $n = 9$).

the language of computer science, information theory, and statistical physics. Our adoption of prediction as a driving principle shifts the focus to how a minimal (yet growingly complex) computation projects memory about the past into the potential future of an environment.

II. METHODS

A. Minimal bit-guesser model

Bit-guessers G are abstract machines that posses a model Γ_E^G of an external environment E. An environment consists of a finite tape with ones and zeros, to which guessers access one bit at a time similarly to how Turing machines read their inputs—only, our guessers advance always in the same direction. Bit-guessers use their internal model Γ_E^G (akin to a Turing machine's internal state) to attempt to predict the next environmental bits. Thus, the tape is obviously read but, differently from Turing machines, in the current implementation our guessers do not *print* upon it—i.e., they do not modify their environments. In the following we introduce some notation about each of these elements. (See Ref. [38] for a more thorough description.)

An m-environment ($E \equiv \{e_i;\ i = 1, \dots, m\}$) consists of m bits drawn randomly and uniformly. We evaluate an n-guesser in a given m-environment (with $n < m$) by dropping it at a random position, i_0, and asking that it predicts the next n consecutive bits (Fig. 1) with cyclic boundary conditions. These constitute an n-sized word $B(i_0) \subset E$. The guesser's internal generative model Γ_E^G produces yet another n-sized word, $W_E^G(i_0)$, based on the guesser's memory and its past interaction with the environment (i.e., its history of correct and incorrect guesses as explained below). The fraction of bits correctly guessed reads

$$p_E^G(i_0) = \frac{1}{n}\sum_{i=1}^{n} \delta(w_i, b_i). \tag{1}$$

Here, we use Kronecker's δ: $\delta(w_i, b_i) = 1$ if $w_i = b_i$ and $\delta(w_i, b_i) = 0$ otherwise; and w_i and b_i are the ith bit within $W_E^G(i_0)$ and $B(i_0)$, respectively. We might want to evaluate a guesser several times in a same environment (for which we drop it at different starting positions i_0) or in an ensemble of environments of fixed size m. For simplicity, we use p_m^n to name the average number of bits guessed by an n-guesser in either m-environmental setup.

Bit-guessers pay a cost c for each bit that they attempt to predict, and they rip off a reward r for each correct prediction (Fig. 1). Evaluating a guesser in an environment reports a net reward:

$$\Delta\rho = \left(p_m^n r - c\right)n = \left(p_m^n - \alpha\right)rn, \tag{2}$$

with $\alpha \equiv c/r$ measuring how meager an environment is: the larger α, the less reward per correct guess. If $p_m^n < \alpha$, then $\Delta\rho < 0$ and the given n-guesser could not survive in that m-environment. In some of the numerical experiments that follow, if a guesser's accumulated reward ρ is large enough, then it gets replicated.

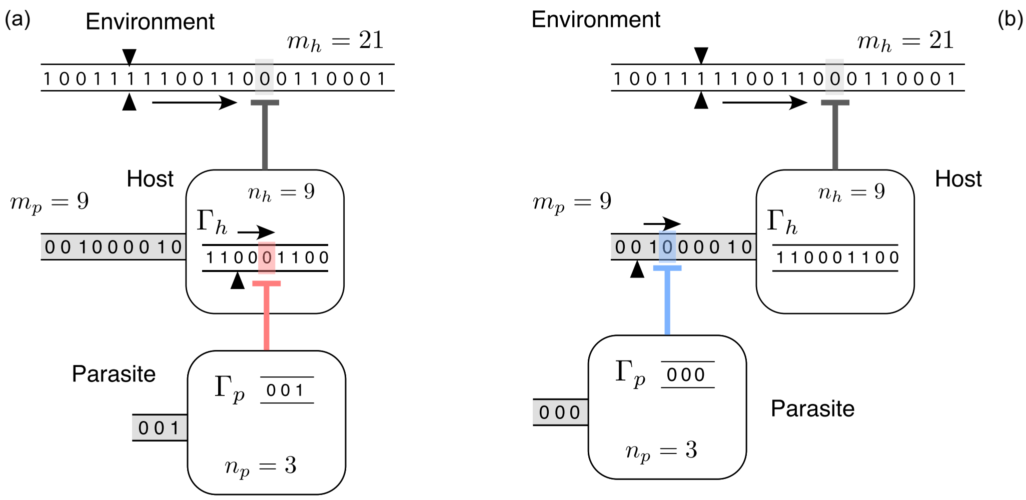

FIG. 2. Host-parasite systems modeled as bit-guesser machines. (a) A genotype parasite uses the host's internal model (Γ_h) as the environment off of which it extracts information to live. Note the reading head (light red) of the parasite interacting directly with the host's internal model about the external environment. (b) Phenotype parasites live out of anticipating the host's behavior—which is different from its internal model of the world. Note the reading head (light blue now) of the parasite now reacting to the external behavior that the host produces, while it does not interact with the host's internal model directly. Both host-based environments (its internal model and its behavior) have the same size, but phenotype parasites will usually encounter less complexity than genotype parasites. This is so due to the model's dynamics: A guesser's behavior is produced from its genotype by resetting each time that a mistake is made, thus making behavior more redundant than the model on which it is based.

Our guessers were designed to have their computational complexity controlled by a single parameter—its size, n, which also imposes a replication cost. This computational complexity is implemented by the guesser's generative model, Γ_E^G, which consists of an n-sized bit-string used sequentially for predicting E. To elaborate Γ_E^G, we assume that the guesser has had access to the whole environment, and that it has come up with the best model possible constrained only by its size n and a minimal correcting mechanism. The only information unknown to the guesser is where it will be dropped as it is evaluated later. Thus, $\Gamma_E^G(1)$ is just the most frequent bit in the environment, which is the most likely outcome if we drop it in a random position. After a correct guess of this first bit, moving onto the next position, the most likely guess $\Gamma_E^G(2)$ is the most frequent bit following every instance of $\Gamma_E^G(1)$ in the environment. We proceed building $\Gamma_E^G(i)$ as the most frequent bit following all $(i-1)$-sized words matching $\{\Gamma_E^G(1), \dots, \Gamma_E^G(i-1)\}$ in the environment.

As we evaluate a guesser, it proposes consecutively the bits in Γ_E^G as long as its predictions are correct [Fig. 1(a)]. If there is a mistake [Figs. 1(b) and 1(c)], then the guesser resets to $\Gamma_E^G(1)$ for the next bit and proceeds onwards from there. More formally, the word $W_E^G(i_0;k)$ produced by guesser G in the environment E [Fig. 1(d)] when it is dropped in i_0 is $W \equiv \{W(k), W(k) = \Gamma_E^G(k-l)\}$ where $k = 1, \dots, n$ and l is the last k such that $W_E^G(i_0;k) \neq B(i_0 - 1 + k)$. $l = 0$ if no wrong guesses have occurred yet.

B. Genotype and phenotype parasites

Our model makes a distinction between the ideal best guess Γ_E^G and the bits actually emitted as a guesser is evaluated W_E^G. We liken these to genotypes and phenotypes. Γ_E^G stores instructions for prediction and dictates the agent's behavior expressed by W_E^G. Parasites making a living off of another organism can do so by predicting the host's inner structure or its behavior. In the real world, gene matching as well as external trait recognition can be used by parasites to recognize their targets. To consider these two scenarios, we study both and label as *genotype* and *phenotype parasites*, respectively. The former take the host's Γ_E^G as their environment [Fig. 2(a)], while the later dwell on the host's external features (such as behavior) W_E^G [Fig. 2(b)]. Because bit-guessers distill correlations from their environments, both Γ_E^G and W_E^G are more predictable than completely random environments of the same size. It is simpler to predict Γ_E^G and W_E^G, but in turn they shall provide less reward.

We now need to differentiate between (i) the complexity of the external (or host's) environment (m_h) and the host's complexity ($n_h < m_h$); and (ii) the complexity of the parasite's environment, which is always the host's size ($m_p = n_h$), and the parasite's complexity ($n_p < m_p$). For clarity, we omit the host and parasite subindexes (h and p), and just put a bar over all variables referring to parasites. Thus, we name: $m \equiv m_h$, $n \equiv n_h$ (which equals $\bar{m} \equiv m_p$), and $\bar{n} \equiv n_p$. Evaluating parasites as usual, we have

$$\overline{\Delta\rho} = \left(\bar{p}_{\bar{m}}^{\bar{n}} - \bar{\alpha}\right)\bar{r}\bar{n}. \tag{3}$$

Note $\bar{\alpha} \equiv \bar{c}/\bar{r}$ controlling the reward per correct prediction for parasites, with $\bar{\alpha} \neq \alpha$ in general. This parasite reward is subtracted from the host, so Eq. (2) is modified into

$$\Delta\rho = \left(p_m^n - \alpha\right)rn - \bar{p}_{\bar{m}}^{\bar{n}}\bar{r}\bar{n}. \tag{4}$$

We assume that all the reward taken away from the host is efficiently transferred to the parasite. Variations allowing leaks would not affect our results substantially.

III. RESULTS

A. Bit-guesser size versus other measurements of complexity

Complexity is a thorny concept that eludes a clear-cut definition—we can find as many as authors [41]. Many see this as a drawback for complexity science; but a similar caveat can be found regarding the definition of "life" and "living," and this has not prevented the advancement of biology—including in borderline cases such as the study of viruses. Besides having sensible measurements of complexity, we need to know how they relate to each other, understand what they are telling us mathematically, and assume that a same work can be seen from different angles.

Our model takes the sizes of environments and guessers as a measurement of their complexity. In the case of environments, this seems straightforwardly correct because they consists of periodically repeating strings of random bits. The longer an environment, the more variety and diversity we may encounter. In the case of guessers, we argue that their size is a good measurement of complexity because each additional bit confers them an expanded ability to find patterns within the environment. We made a deliberate choice to place all the guesser's algorithmic capability in its memory, specifically to have a single number comparable across guessers controlling complexity. Alternative implementations of guessers will result in different ways of quantifying procedural complexity, but similar results should follow as long as appropriate costs are placed on the expanded computational capabilities. One such example would be a guesser that, instead of one pattern (as in ours), stores a decision tree that is navigated whenever mistakes are made. In such a case, a measure of complexity should include the number of bits (like ours) but also the different branches and their depth, as well as placing costs on each decision.

Information theory offers ways to formalize these concepts, but not a straightforward manner of assigning costs to the different ingredients. Kolmogorov complexity [42] is the length of the shortest program that can produce a given binary string. This measure is maximal for truly random strings, as the impossibility to predict the next bit from the previous history implies that any program producing the string must contain it in full. Completely regular strings (say, one that repeats "01" *ad infinitum*) have minimal complexity. In general, the Kolmogorov complexity of a string cannot be computed. Different correlates are used to approximate it in practical situations—for example, some notions of entropy calculated over the probability of finding different patterns within a string [22,43]. Compression methods attempt to find minimal representations of a given string, thus they often approximate Kolmogorov complexity as well. The lossless Lempel-Ziv-Welch (LZW) algorithm [44,45] explicitly finds repeated patterns of variable length within a longer string to remove unnecessary redundancies.

Figure 3(a) shows the Kolmogorov complexity (as approximated by the LZW algorithm) for environments of increasing size. As expected, this grows monotonically with size—as m grows, we encounter longer strings of random bits which require more and longer unique patterns to reconstruct. Thus, environment size, m, is a good correlate of its Kolmogorov complexity. Figure 3(b) shows that guesser size, n, is also a good correlate of its corresponding complexity. This complexity seems to grow linearly with n, but it displays a much larger variance than environments. This suggests that the relationship between a guesser's size and its Kolmogorov complexity can be highly nonlinear and context dependent.

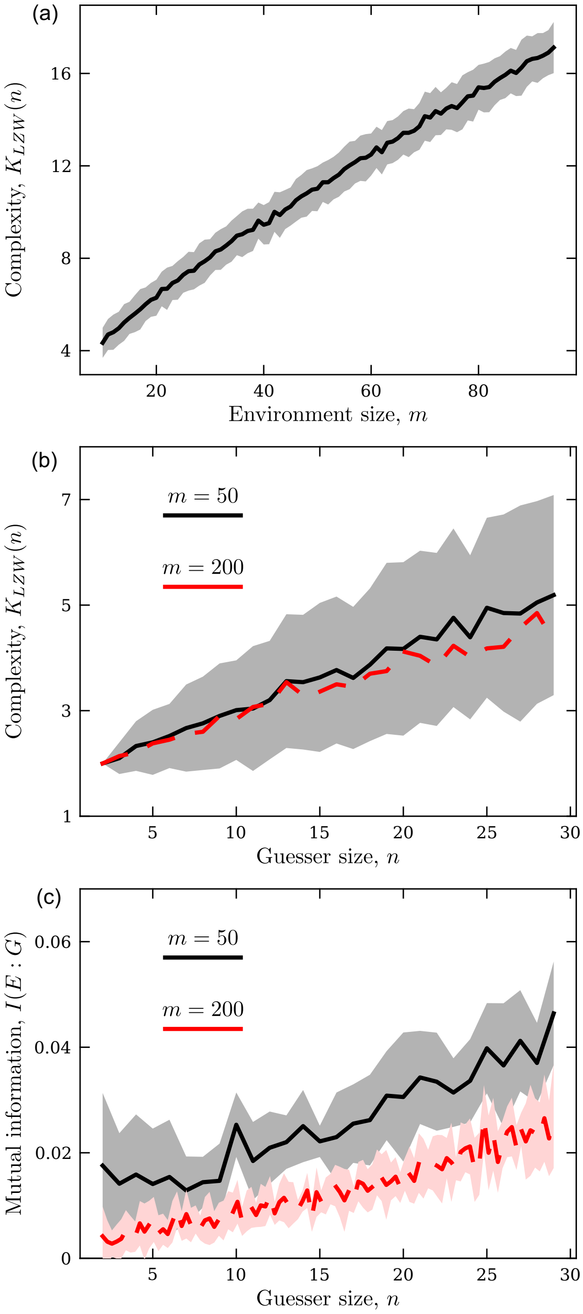

FIG. 3. Comparing measurements of complexity. (a) Environment complexity as a function of its size, m. Average over 100 environments of each size, shading is standard deviation. (b) Same for guesser output as a function of its size, m. One evaluation in each environment with $m = 50$ (black) and 200 (red, dashed; shading omitted). (c) Mutual information between environments and guesses. Ten repeats of one evaluation of the guesser in each of 100 environments.

The crucial ability of a guesser to thrive comes from its capacity to correlate its bits with those of the environment,

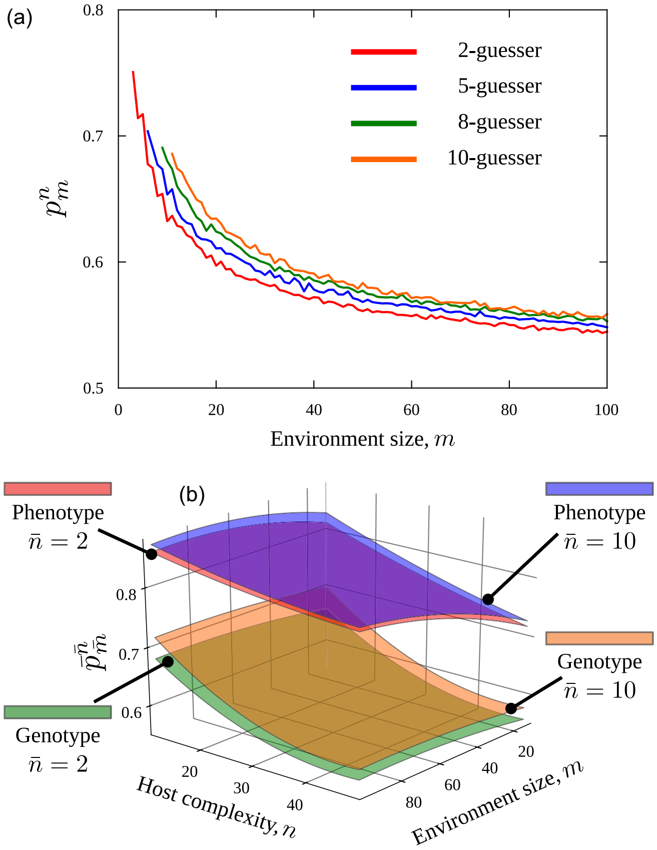

FIG. 4. Host and parasite performances. (a) Host's performance (p_m^n) as a function of its complexity and the environment's size. Curves are for 2-, 5-, 8-, and 10-guessers, with curves for bigger guessers laying on top and shifted to the right with respect to those for smaller guessers. (b) Parasite's performance ($\bar{p}_{\bar{m}}^{\bar{n}}$) as a function of both host's complexity and external environment's size. Shown are least square fits of quadratic surfaces to data from numerical experiments.

a property that we can capture with information theoretical and statistical measurements such as mutual information. In Ref. [38], we showed how this decreases, for guessers of a fixed size, as environments grow larger; and how it grows, for fixed environment sizes, as a guesser's complexity increases. Figure 3(c) shows an updated view of this relevant information-theoretical measure.

B. Performance in environments of different sizes

Our simplest question is how many bits can hosts and parasites guess under specific circumstances. Expanding results from Ref. [38], Fig. 4(a) shows p_m^n for hosts. These curves delimit average survival conditions: n-guessers can survive in (m, α) combinations under their corresponding curve. More complex guessers (larger n) can survive in a wider range of conditions. For parasites, we vary both external environment (m) and host complexity ($n \equiv \bar{m}$). The volumes subtended by the surfaces in Fig. 4(b) indicate combinations $(m, n = \bar{m}, \bar{\alpha})$ under which $\bar{n}$-parasites can survive. This volume is again bigger for more complex (larger $\bar{n}$) parasites. Survival regions decrease as host complexity grows (Fig. S1(a) [46]), hence becoming more complex is a good strategy to diminish the survival chances of parasites.

Survival surfaces associated to phenotype parasites subtend much larger volumes than those of genotype parasites in Fig. 4. Due to the reset mechanism in generating W_E^G, its complexity is bounded by that of Γ_E^G, thus making W_E^G more predictable. Here our model differs from real organisms, in which genotypes are seeds that generate much more complex behavior. In any case, both W_E^G and Γ_E^G are easier to predict than external environments of the same size, which is the crucial advantage of parasitism.

C. Escaping parasites by increasing behavioral complexity

A straightforward way to become more unpredictable is by adding some randomness (hence, computational complexity, as rigorously defined [42,47–49]) to a planned behavior. We implemented hosts with their usual Γ_E^G, but who flip each bit in W_E^G with probability p_F. Figure 5(a) shows how the net reward changes for hosts ($n = 10$) and parasites ($\bar{n} = 4$) as p_F varies. The effect depends on the environment size (here $m = 30$) and the parasite's share of reward ($\bar{r} = 0.5$). In Fig. 5(a), when p_F is low, parasites survive, resulting in less reward for the host. For large enough p_F, the parasite is unable to predict its host and dies. But the randomness introduced entails that the host fails to predict some of the bits that it could, thus very large p_F has a toll eventually. A tradeoff emerges between the parasite and environmental pressures resulting in an optimal level of randomness [$\hat{p}_F$, Fig. 5(b)] for the host. The host's reward at this optimal value [$\hat{\rho}$, Fig. 5(c)] can be diminished by either factor.

For $0 < p_F < \hat{p}_F$ the strategy partly thwarts the parasite's development. This would result in a slowed-down epidemiological spread. At the level modeled here, arrested (yet surviving) parasites have a smooth, parsimonious effect on the host. This regime might extend to $p_F > \hat{p}_F$ values, as $\hat{p}_F$ marks the threshold at which further random behavior by the host does not pay off, which is compatible with residually surviving parasites.

If parasites take away very little reward (e.g., $\bar{r} = 0.4$, Fig. S3(a) [46]), they cannot survive even in normally behaving hosts; thus $\hat{p}_F = 0$. This regime persists for a range $\bar{r} < \bar{r}^*$ [Fig. 5(b), shaded area]. If $\bar{r}$ is very large (e.g., $\bar{r} = 0.7$, Fig. S3(d) [46]), parasites survive even in fully unpredictable hosts. Thus, flipping bits does not help the host against the parasite and still results in worst prediction of the external environment—hence $\hat{p}_F = 0$ again. Only $\hat{p}_F > 0$ for an intermediate range of $\bar{r}$ (Figs. S3(b) and S3(c) [46]). Even if parasites are driven to extinction in this regime, the host misses some of the potential reward due to its degraded ability to predict [Fig. 5(c)].

D. Increase of complexity in neutral ecosystem dynamics

We model neutral ecological interactions [Figs. 9(a)–9(c)] with ecosystems that contain a number of spots. Each spot can be occupied by an n-guesser (restricted to $n = 1, \ldots, 10$ in the next example) or empty (a 0-guesser). An ecosystem presents fixed environment size (m) and harshness (α). All guessers are initially endowed with a reward $\rho(t = 0) = n\rho_0$, representing a satisfied metabolic load that grows with the guesser's complexity.

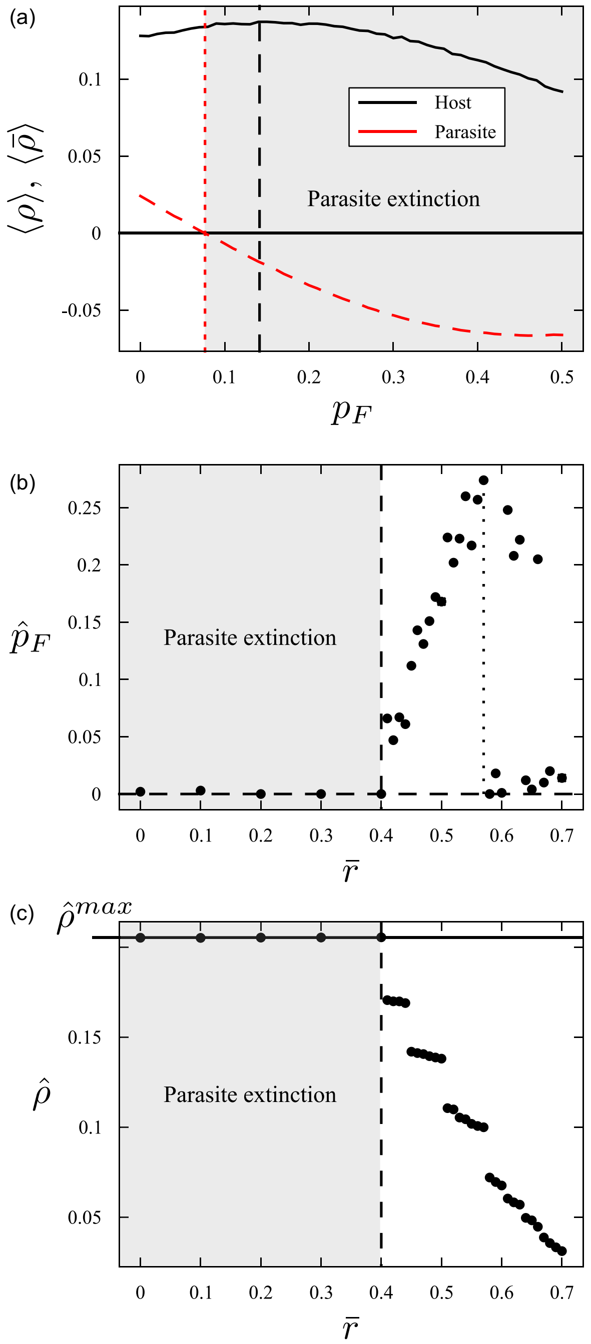

FIG. 5. Escaping parasites by increasing behavioral complexity. (a) Host ($n = 10$, black solid curve) and parasite ($\bar{n} = 4$, red dashed curve) average reward for $p_F \in [0, 0.5]$ ($m = 30$, $\bar{r} = 0.5$). For each $\bar{r}$, we evaluate 10^5 hosts and parasites. If the parasite survives, then $\bar{p}_m^{\bar{n}} \bar{r} \bar{n}$ is subtracted from its host's reward. Optimal flip probability $\hat{p}_F$ (b) and reward $\hat{\rho}$ (c) reveal three different regimes: (i) Low $\bar{r}$ (shadowed area), parasites die naturally, no additional behavioral complexity is needed ($\hat{p}_F = 0$, $\hat{\rho} = \hat{\rho}^{\max}$). (ii) Intermediate $\bar{r}$, some additional behavioral complexity eliminates parasites ($\hat{p}_F > 0$, which has a toll so that $\hat{\rho} < \hat{\rho}^{\max}$). (iii) Large $\bar{r}$, parasites endure, additional complexity does not help ($\hat{p}_F = 0$ again, $\hat{\rho} \ll \hat{\rho}^{\max}$ because of the lasting parasite).

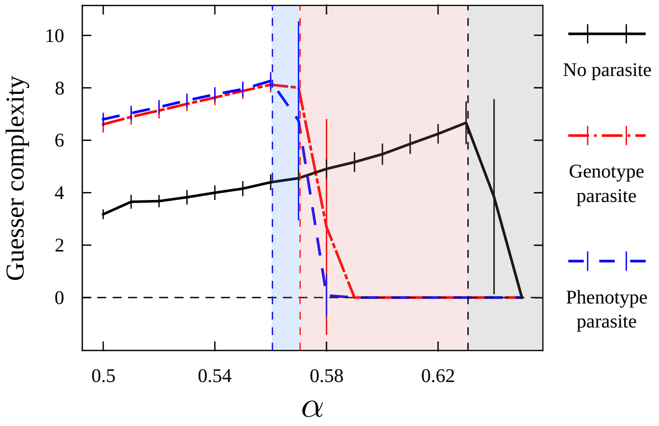

FIG. 6. Ecosystem dynamics. Average complexity of surviving guessers in an ecosystem after 200 generations for a range of $\alpha \in [0.5, 0.65]$ for experiments without a parasite (black, solid line) and for ecosystems with genotype (red, dot-dashed) and phenotype (blue, dashed) parasites. Shaded areas mark ecosystem collapse as environments become too challenging for all guessers present. Parasites always have $\bar{n} = 1$ in this example. The effect of such simple parasite is huge. Note, however, that we are exploring precisely the region of parameter α that separates persistence from extinction for the studied guessers [as seen in Fig. 4(a) for $m = 30$].

To simulate ecosystem evolution, at every time step a spot is randomly chosen. The corresponding guesser is evaluated on a newly generated m-environment [Fig. 9(a)]. As before, evaluation starts after the guesser has elaborated the best internal model (Γ) given its capabilities (n); then it is dropped on a random position of the environment and proceeds as in Fig. 1. The balance from this evaluation is added to the guesser's accrued reward:

$$\rho(t + \Delta t) = \rho(t) + \left(p_m^n - \alpha\right) r n. \tag{5}$$

If $\rho(t + \Delta t) > 2n\rho_0$, then the guesser gets replicated [Fig. 9(b)] and an amount $n\rho_0$ (as initial endowment for the daughter) is subtracted:

$$\rho(t + \Delta t + \delta t) = \rho(t + \Delta t) - n\rho_0. \tag{6}$$

The guesser keeps replicating until $\rho(t + \Delta t + \delta t) < 2n\rho_0$. Daughters are allocated to random spots in the ecosystem, which might be empty or not—in which case the older guesser is replaced. If, after being evaluated, $\rho(t + \Delta t) < 0$, then the selected guesser dies and a 0-guesser occupies its spot [Fig. 9(c)].

We evolved ecosystems with 1000 spots, initially populated by random, uniformly distributed guessers with $n \in [1, \ldots, 10]$. Figure 6 (black, solid line) shows the average guesser complexity after 200 generations (each generation consists in the sequential evaluation of 1000 spots chosen at random, so in average every spot is evaluated once per generation) for $m = 30$ and $\alpha \in [0.5, 0.65]$. In copious environments (low α, large reward per correct guess) a few right predictions report large benefits. Simpler guessers replicate faster because of their lower replication threshold (proportional to $n\rho_0$). If a cheap, sloppy strategy provides enough reward, then investing in costly computations is unfavored. But in austere environments (larger α) simple guessers cannot accrue reward fast enough and a complex computational machinery pays off. Very meager environments ($\alpha \to 1$) become prohibitive for all guessers—as indicated by the sudden fall of complexity in Fig. 6.

To each guesser, we now add a parasite that does not die nor replicate. It just sucks off reward according to Eq. (4) whenever its host is evaluated. Even the simplest ($\bar{n} = 1$) parasite, either at the genotype (Fig. 6, red dash-dotted line) or phenotype (blue dashed line), suffices to achieve a remarkable complexity boost. Simpler hosts are more predictable, hence parasites extract more reward—favoring complex hosts. In return, the point at which the ecosystem collapses happens for lower α. Parasites are double-edged swords that drive complexity but can precipitate their host's extinction. This collapse happens slightly earlier for phenotype parasites—as expected, since, as discussed above, the complexity (and hence predictability) of W_E^G is bounded by that of Γ_E^G.

E. Open-ended Red Queen dynamics

To study parasitic pressures in an ecoevolutionary setup we add a shadow ecosystem where parasites of different complexity dwell and undergo replication and death dynamics (Fig. 10). Replicating hosts and parasites can now produce simpler or more complex daughters through mutations.

All guessers replicate as explained above if

$$2n\rho_0 \leqslant \rho(t + \Delta t) < (2n + 1)\rho_0. \tag{7}$$

Moreover, if the inequality

$$\rho(t + \Delta t) \geqslant (2n + 1)\rho_0 \tag{8}$$

holds, then a mutation happens with probability $p_\mu = 0.5$. (This relatively high mutation rate was chosen to illustrate model dynamics within a reasonable time and without using too many computational resources. Similar results as follow are obtained with smaller mutation rates, but the interesting dynamics that we report below take longer—thus consuming much more computing time.) This mutation goes in either direction ($n \rightarrow n + 1$ or $n \rightarrow n - 1$) with equal chance. The corresponding initial endowment for the daughter is subtracted from the mother. When a host gets replicated, its parasite (if any) gets replicated alongside and occupies the corresponding spot in the shadow ecosystem (replacing an older parasite if necessary). The initial endowment of this daughter parasite is subtracted from the daughter host. Note that parasites also replicate through the usual route—i.e., their reward overcoming a threshold

$$\bar{\rho}(t + \Delta t) > 2\bar{n}\bar{\rho}_0. \tag{9}$$

When this happens, the daughter parasite substitutes the guesser in a random spot of the shadow ecosystem. Parasites might also die, leaving a spot of the shadow ecosystem unoccupied. Thus, a host's parasite might change over time or disappear. The current model assumes $\bar{n} < \bar{m} \equiv n$, thus if a daughter parasite becomes too big (e.g., because it is allocated to a spot with a small host or with no host at all), then she is promoted to host into the main ecosystem.

Despite the simplicity of the model, it resulted in an unexpected range of behaviors—of which we only show the most salient ones. For certain parameters, the host-parasite ecoevolutionary interactions ignite Red Queen dynamics [50] in which both counterparts engage in a race of growing complexity, potentially without an end point, suggesting open-ended evolution [Figs. 7(a)–7(c)]. All simulations were chosen such that, in the absence of parasites, the host ecosystem would be fully populated by 1-guessers. This is, the observed trajectories are truly outstanding. Furthermore, 100 repeats of each experiment have been carried out for a broad range of model parameters, robustly finding that Red Queen dynamics ensue sooner or later for many parameter settings (Fig. S4 [46] illustrates complexity evolution for the 100 experiment repeats with different parameters settings).

Figure 7(d) illustrates the mechanism behind the emergence of these Red Queen dynamics. Simpler hosts are easy to predict, while more complex ones escape their parasites. However, more complex parasites promptly thrive, while simpler ones fail to accrue reward and replicate. This results in a pair of effective forces pushing hosts and parasites toward ever-increasing complexity. We halted our simulations if n became larger than m, but our experiments do not show signs of slowing down. There is no principled reason why this mechanism could not operate indefinitely.

Our simulations show other noteworthy, unexpected phenomena. We observed long periods of stasis [Figs. 7(a)–7(c)], suggesting relatively stable attractors which, anyway, are eventually escaped. Genotype parasites [Figs. 7(a) and 7(b)] present more diverse temporal trajectories. Some show uninterrupted, relatively monotonous complexity buildup [Fig. 7(a)]. Others show ecosystem-wide complexity collapses [Fig. 7(b)] into an alternative, seemingly metastable attractor with complex hosts and simple parasites. These states can persist for hundreds of generations, but are eventually escaped. The time trajectories of phenotype parasites [Fig. 7(c)] are less diverse. In Fig. S5 [46] we show how a range of behaviors can be achieved by varying just one parameter. Some of these dynamics are outstanding—e.g., punctuated equilibrium for very low $\bar{\rho}_0$. These behaviors were never purposefully hardwired into our minimalist model.

The complexity gap between hosts and parasites grows linearly during Red Queen dynamics [Fig. 7(f)]. Perhaps host and parasite complexity must observe some algebraic relationship to enable open-ended evolution—e.g., if hosts get too complex, then they might kill off all parasites and halt the open-ended dynamics.

Notice that fluctuations in complexity grow as all guessers become more complex. This happens both as fluctuations of the average population complexity [and most notably for phenotypic parasites, as Fig. 7(g) illustrates], and as spread within the population at each time [Fig. 7(g)]. This panel shows the standard deviation of guesser sizes within the ecosystem at each time. We observe that this variance is larger for parasites during the stale phase, but that it becomes larger for hosts during the Red Queen dynamics. Both hosts and parasites present: (i) large fluctuations in the variance of their complexity and (ii) an overall trend of increasing standard deviation of complexity. The fluctuations indicate large, sudden contractions and expansions of the distribution of guesser sizes, suggesting nontrivial ecoevolutionary interactions. The overall increasing standard deviation of the complexity, however, indicates that not only are both hosts and parasites becoming more complex over time, in average; but also both populations themselves are becoming more heterogeneous over time. This

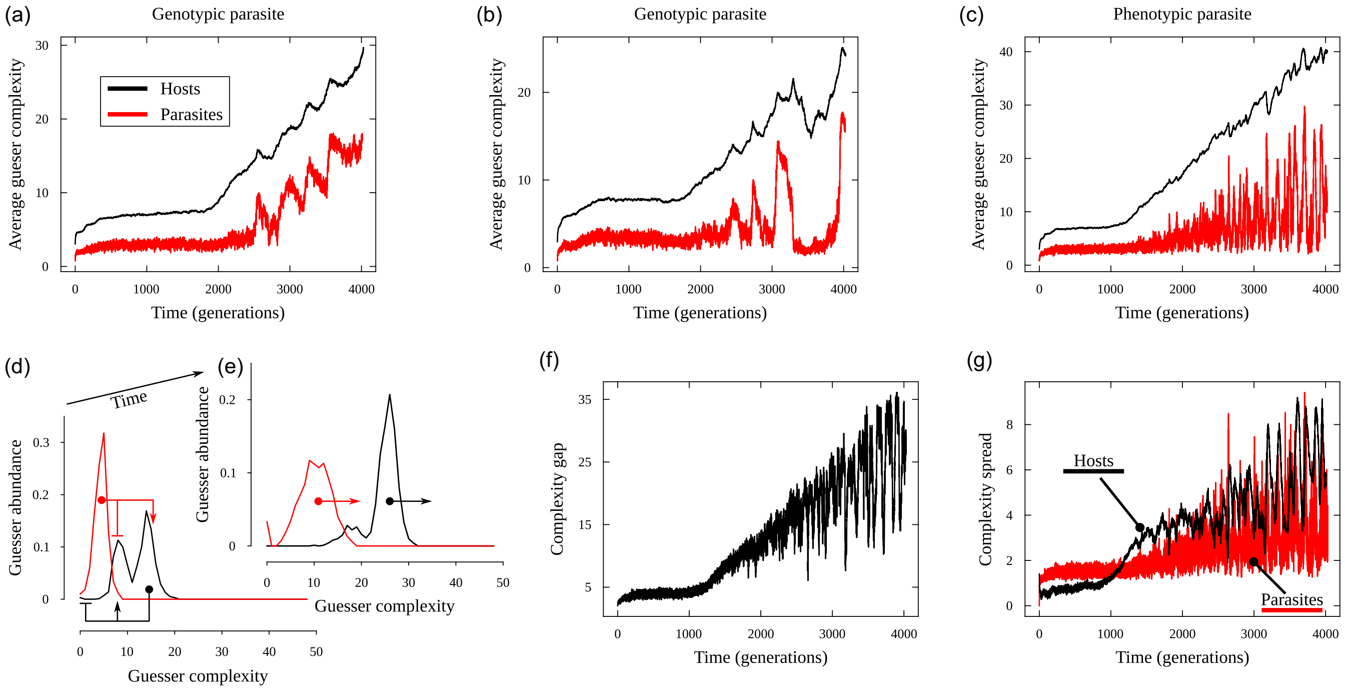

FIG. 7. Red Queen dynamics in host-parasite coevolution. (a)–(c) Single run of a time evolution of average complexity of hosts (black, top curve) and parasites (red, bottom curve) (ecosystems have 1000 spots, $m = 50$, $\alpha = 0.5$, $r = 1$, $\rho_0 = 1$, $\bar{\alpha} = 0.6$, $\bar{r} = 0.35$, $\bar{\rho}_0 = 0.005$, and $p_\mu = 0.5$). Genotype parasites (a), (b) present a more varied temporal unfolding than phenotype parasites (c). (d), (e) Abundance of guessers of different complexity between hosts (black, right-most distributions) and parasites (red, left-most distributions) at two different times. Simpler hosts are easier to predict by their parasites, more complex hosts get away. Simpler parasites fail to cope with their hosts, while more complex parasites fare better. Effectively, simpler guessers are repressed and more complex ones are favored. (e) This results in a pair of effective forces pushing both host and parasite communities toward ever-higher complexity. (f) Average complexity gap between hosts and parasites of the simulation in panel (c). (g) Standard deviation of complexity of host (black) and parasite (red) populations at each given time of the simulation in panel (c)—not to confuse with the wide variation of the average population complexity over time.

suggests that, as the Red Queen dynamics proceed, more combinations of host-parasite couples might become viable.

Finally, we could think that more complex parasites might be lethal and result in emptied ecosystems (as it happens without evolutionary dynamics for large α, Fig. 6). Figure S6 [46] shows, instead, that ecosystem occupation remains at its maximum for hosts and increases lightly over time for parasites.

IV. DISCUSSION

It has been conjectured that parasites act as a pressure for more complex hosts. Here we sought the simplest mathematical description that captures this qualitative hypothesis. We built upon the bit-guesser model [38] as a minimal framework in which complexity (as grounded in information theory and computer science) is parsimoniously connected to Darwinian selection by organisms that thrive and replicate if they successfully predict their environment. These elements (replication, selection, and information processing), among others, set biology apart from inert matter [1–5].

Our minimalist model shows that: (i) increased behavioral complexity can be a valid strategy to scape parasites; (ii) more complex organisms result easily from the introduction of simple, immutable parasites in an ecosystem; and (iii) ecoevolutionary dynamics can result in Red Queen dynamics of hosts and parasites becoming more complex to scape each other. These results join other explicit drivers of increased biological complexity [38], as well as evidence from large census of species [Fig. 8(c)] [51]. Together, they weaken more neutralist views [23] which propose that there are not explicit evolutionary pressures favoring higher complexity in biology. It is reasonable to think that these forces (notably parasitism) have been operating since very early in the history of life. This offers reassuring arguments that complex life is expected under an array of circumstances, increasing the likelihood that organisms such as higher metazoans or advanced cognitive systems did not arise by random drift.

The emergence of Red Queen dynamics is our most important finding. A wide set of conditions lead to a quick, seemingly open-ended, surge of complexity. Our results imply that powerful forces underlying common biological interactions will drive life toward great complexity under the adequate conditions. In our opinion, this robust mechanism turns the original question ("Are there pressures toward complex life in Darwinism?") on its head. We should now wonder what the consequences of this strong evolutionary dynamics can be in the real world, or under what circumstances this mechanism can be attenuated or harnessed. In this last regard, we say nothing about the explicit implementation through which this complexification can be achieved.

Our model is limited by finite size constraints, but the complexity increase did not show a tendency to stop or saturate. There is no reason why the principles operating at

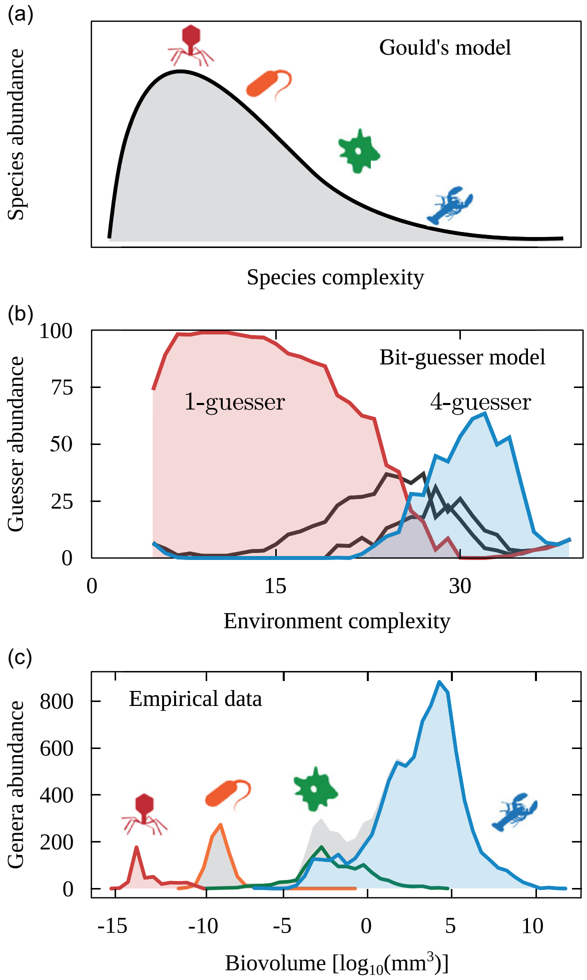

FIG. 8. Different models of organismal complexity across scales. (a) A proposal by Gould [23] that more complex life arises through sheer random drift leads to a unimodal distribution of life complexity peaked around the *most successful* class of organisms (bacteria, according to Ref. [23]). This is a cartoon distribution modified from Refs. [23,51]. If evolutionary pressures for the emergence of complex life exist, then it becomes more likely that steps of organismal complexity occupy their niches and present multipeaked distributions across scales. Some such evolutionary drivers were numerically characterized in Ref. [38] and produce multipeaked distributions ((b), where abundances show percentage of ecosystem occupation as in Fig. S6 [46]). Multipeaked distributions are also derived from exhaustive data bases [51] (c).

the studied scales should not work indefinitely, suggesting that host-parasite dynamics might drive open-ended evolution. This feature can be suggested from our model, that shows much longer and more sustained increases in complexity than other models [35,52] and it occurs at seemingly constant rates. This suggests that complexity does not affect much the rate of evolution. Even within the limited range studied, a very rich array of behaviors emerge. Noteworthy are the observed collapses of complexity [Fig. 7(b)], which suggests that forward complexity evolution requires hosts and parasites progressing apace; and the punctuated equilibrium observed for very small parasite replication threshold ($\bar{\rho}_0$, Figs. S5(e) and S5(f) [46]). These features were not anticipated. The minimalism of our model suggests that such phenomena might be general within host-parasite coevolution. The fact that the collapses do not always happen [Fig. 7(a)] and that they can be overcome [Fig. 7(b)] shows that they are not an upper limit of complexity (which does not prevent it from existing anyway).

Multiple biological examples of host-parasite coevolution [53–81] (notably those involving genomes) present elements very reminiscent of computer science and information theory (e.g., operations based on pattern matching, copying, etc.). Thus, despite the abstraction of our model and results, it is not unreasonable to try comparing them to empirical data. This paper is a first step in that direction. We derived quantitative bounds to computational aspects of host-parasite interactions, similar, e.g., to thermodynamic bounds for computation that apply in physical and biological systems [82]. Such bounds are better embodied by the $\bar{p}_{\bar{m}}^{\bar{n}}$ surfaces in Fig. 4(b). Further qualitative observations relevant for potential empirical studies are the gaps observed between host and parasite complexities. Might this gap also grow linearly in real host-parasite systems? To asses this we need to tackle complexity in real biological systems—a difficult issue [83]. Genome size is a first proxy, but examples of simple organisms with large genomes abound. The Kolmogorov complexity [42] of a genome might remove redundancies and get us closer to the amount of useful information encoded. Beyond this, protein, functional, and behavioral repertoires might be our next approximations to capture the complexity of living beings.

Here we focused on a specific, biologically inspired view of parasitism. However, our mathematics demand only two populations of coevolving agents with one of them making a living out of predicting the other, upon which a small damage is inflicted. The phenomenology that we uncover should be relevant for any situation fulfilling these conditions. Within biology, certain aspects of male-female interactions have been framed as host-parasitic relationships, with males of some species (notably among fish) openly described as parasitic. Our results would imply that the split in two sexes could, in certain cases, be yet another powerful engine for fast biological complexification. However, additional aspects should be taken into account (e.g., shared descent between hosts and parasites) which might modify the dynamics.

An important feat in the early evolution of life is the transition to a code capable of Darwinian evolution (and the computational processes that this demands) [84,85]. This acquires a greater importance in metabolism-first models of the emergence of life. Before an actual code exists, such models often rely on autocatalytic cycles as a self-replicating structure with Darwinian dynamics, without a centralized encoding of the information. What might have prompted the computational complexification of these structures in the first place? It turns out that autocatalytic cycles are affected by their own parasites as well [86–88]. These are molecules or metabolic pathways that benefit from the catalytic activity without implementing any step of the cycle (i.e., they are metabolic drains). Under the light of our results, such models would be equipped with a ready-made evolutionary pressure for increasing computational complexity from the very beginning. The emergence of new parasites would be dictated by the exploration of new metabolic pathways by the autocatalytic set itself. Phenomena such as the periods of stasis that we observe would provide ratcheting platforms that would prevent mounting complexity from fading away.

The essence of our framework might capture nonstandard parasites in social, technological, or economic systems (e.g., as businesses profit from each other's intellectual efforts, as hackers exploit the computational power of unaware remote servers [89], or as traders anticipate each other's moves [90]). Parasites might also have shaped the evolution of neural structures [91], thus extending our questions to machine learning and cognition in general. The recent, outstanding success of generative adversarial networks [92,93] relies on two systems (while of fixed complexity) establishing antagonistic dynamics similar to ours: a network gains fitness by fooling the other with artificial data, and the other becomes fitter by learning to discern fabricated examples. Our results offer a window to study the emergence of increasingly complex representations in digital ecosystems, as well as serious hypotheses about drivers of advanced cognition in the real world.

ACKNOWLEDGMENTS

We thank members of the Complex Systems Lab and Dr. Amor from the Gore laboratory at MIT's Physics of Living Systems for useful discussion. Seoane acknowledges the important insights contributed by all members of RCC. This work has been supported by the Botín-Foundation by Banco Santander through its Santander Universities Global Division, MINECO Grant No. FIS2015-67616-P (MINECO/FEDER, UE) fellowship, and AGAUR FI grant by the Universities and Research Secretariat of the Ministry of Business and Knowledge of the Generalitat de Catalunya and the European Social Fund and by the Santa Fe Institute. Seoane was funded by the Spanish Department for Science and Innovation (MICINN) through a Juan de la Cierva Fellowship (Grant No. IJC2018-036694-I).

APPENDIX A: THE BIT-GUESSER MODEL, STEP BY STEP

Our current work builds upon the bit-guesser model introduced in Ref. [38]. This model seeks a minimal representation of self-replicating agents that must predict their environment to survive. In searching for minimalism, some conceptual licences are taken and the model turns rather abstract. This section contains an informal, yet more detailed discussion of the model in the hope that it will facilitate the understanding of its different elements.

1. The environment

Environments in the bit-guesser model consist of a finite tape of size m containing zeros and ones. This is inspired by Turing's original model of computing machines. Within our model we are usually interested in knowing the performance of a bit-guesser in sets of environments that constitute ensembles in a same class of equivalence. For example, we often want to know the average performance of a guesser in the ensemble E^m consisting of all environments with size m. This becomes unrealistic for large enough m, so we are content to sample a subset $\hat{E}^m \subset E^m$. Note that two m-environments are usually different: one might have all 0s and another one a balanced mixture of 1s and 0s. Environments in such an ensemble are usually generated on the spot as needed, and stochastically; thus at least some average properties are equivalent, especially when we evaluate a guesser over many replicas.

Environments have periodic boundary conditions, such that if we move along them, their last bit loops back to the first one. Both this periodicity and the finiteness of our binary tapes are choices to study how much useful information can be extracted from them. Instead of m-environments, we could have modeled an infinite tape with a random distribution of bits; and have this distribution modulated by a parameter that would introduce correlations between positions on the tape. This would result in an unbalance of *words* of different sizes, which would make certain patterns more predictable than others. The finiteness of our m-environments has an effect similar to such correlations. The mechanism is at follows:

(i) In a short environment, we expect to find more deviations from an ergodic sampling of all possible distributions of 1s and 0s than in a larger environment. As m grows, the chance that we generate a random environment with an excess of, say, 0s becomes much smaller. Thus, for larger enough m, the most likely is that both 0s and 1s are equally represented, thus knowing the likelihood of the most frequent bit will be each time less useful to predict any bit in the environment.

(ii) The same would happen if we would look at the distributions of 2-words. This is: in short environments it is more likely that we find an unbalance of 00, 01, 10, or 11. The balance is recovered as m grows. However, since the distribution of 2-words needs more information to be fully specified, an ergodic sampling is reached for higher values of m than for 1-words. This implies that 2-words remain useful over a wider range of m-environments.

(iii) The same happens for distributions of longer words. On top of that, computational mechanisms (such as our bit-guessers) that can exploit the information of words of a given length can usually exploit information of shorter words—thus the benefits accumulate.

There are other ways in which we could have introduced how "meaningful" (and hence worth learning) patters of different sizes are. The finite tapes chosen offer an elegant way that parameterizes environment complexity with just an intuitive number, m.

2. The guessers

Similarly to environments, there are several ways in which we could have modeled self-replicating agents with computational capabilities. We could have opted for recurrent neural networks, epsilon machines, Bayesian or Boltzmann networks, etc. But these options go against our search for minimalism. In conceiving bit-guessers, we were hoping to capture just the essential elements and, hopefully, to summarize a guesser's computational costs and capacities with as few parameters as possible.

We converged to a model that also consists of a tape of bits, plus a minimal "if" to allow for the simplest error correction possible. The internal model of an n-guesser (Γ) consists of n sorted bits, now without boundary conditions. We can conceive an evolutionary dynamics of n-guesser candidates over a given, fixed m-environment: First, let us produce a lot of n-guessers, each with its own model (Γ_i) of the environment generated randomly. Then, let us evaluate each of these guesser candidates (as specified below) in the given, fixed m-environment. Next, let us select the best-performing Γ_i and, from them, produce new guesser candidates through mutation and cross-over of their internal models. As we iterate these steps, we would expect our population to converge to the best guessers possible given their n bits of computational capability. These evolutionary dynamics might be very interesting on their own, but in our research we cared more about the limits to performance that m-environments allow given their own complexity. Therefore, we generated our bit-guessers with arguably the best guess that they could come up bit, given their computational capacity.

Once an n-guesser has been initialized with the best guess possible, it is evaluated by being dropped onto a random position of its m-environment, then requesting that it predicts the bit in that and each of $n - 1$ consecutive positions. Hence, the first best guess possible given the m-environment is the most frequent bit in it. This constitutes the first bit (Γ_1) in Γ. If the guesser is dropped in a position whose bit matches Γ_1, then the best guess for the next position is the bit that most frequently appears after each instance of Γ_1 in the environment. If, from the position where the guesser was dropped, the second bit also matches its model (Γ_2 now), then the best guess for the third position is the bit that most frequently appears after each instance of the 2-word $\{\Gamma_1, \Gamma_2\}$. This goes on, thus these assorted collection of bits (up to the nth iteration) constitutes the best guess with which n-guessers are equipped.

In evaluating a guesser, it produces a behavior (W) that consists of all the bits that it comes up as a guess after it is dropped in a random position of the environment E and advances forward over it. Thus, the first bit in W will invariably be the first bit in Γ as well. If this guess is correct, then the next bit in W will be the next bit in Γ as well, and so on. If one of the guesses is incorrect, then the n-guesser has not got a heuristic to infer *where* it is located within the environment—it got lost. The best it can do is to go back to the starting point, and guess that the next bit will be the most frequent one in the environment—hence the next bit in W is the first bit in Γ again, from which the guesser proceeds as before as long as it keeps proposing correct guesses. The reset mechanism when a guess is mistaken is an ‘if’ command—the only one allowed to bit-guessers.

Two important aspects are worth considering: First, instead of this hard reset, we could have modeled inference machines that navigate a tree. If we would do this, then additional complexity is introduced, as we now need to parametrize the width and depth allowed in the tree. This counters our search for minimalism. It would also not be obvious how to weight the cost of successive “if” statements with respect to memory. Second, when looking at bit-guessers behaving (it helps to write down a few examples), it might occur that the guesser appears “too dumb” not to find an obvious pattern. For example, a 2-guesser could never detect the repeating pattern in the string 110110110110110110. This is, precisely, what it means that the computational capacity of this guesser is limited by $n = 2$.

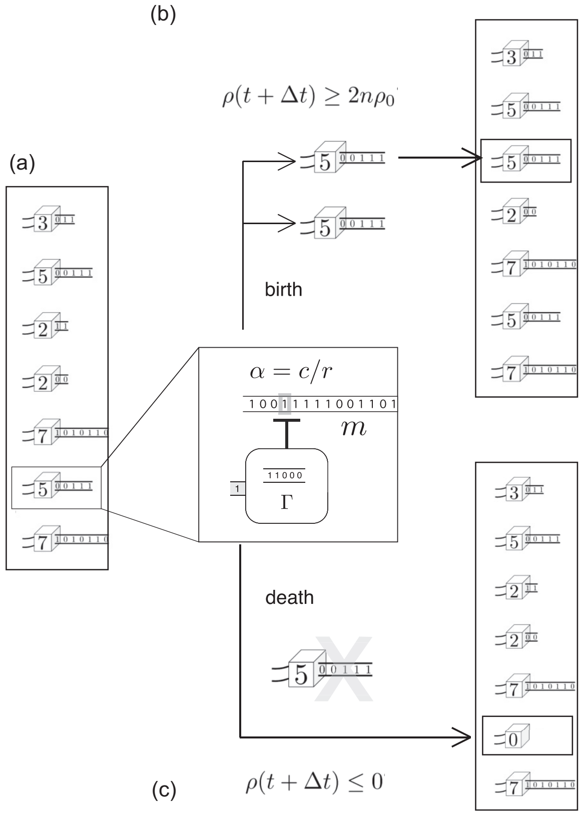

FIG. 9. Ecological dynamics in a simple setup. (a) From ecosystems with a finite number of slots, guessers are randomly picked up for evaluation against environments of a fixed size. (b) Good performance (i.e., good prediction of the environments) can get a guesser replicated. (c) Failing to anticipate the environment can get a guesser killed.

3. Numerical experiments with bit guessers

In Ref. [38] and in this manuscript we describe a series of experiments that involve guessers coexisting or competing in a modeled ecosystem. Our ecosystems consist of a set of N spots, each of which can be occupied, or not, by a guesser. In all our examples $N = 1000$. Ecosystems evolve in *generations*, with one generation consisting on the evaluation (and implementation of death and replication dynamics if necessary) of the guessers in N spots. In all our experiments, ecosystems are identified by environments of a fixed value, m.

The simplest experiment with ecosystems is described in Fig. 9, and corresponds to experiments implemented in Ref. [38]. In this experiment, the ecosystem is initialized by filling in the spots with random guessers that have $n \in [1, \ldots, n^{\max}]$, were $n^{\max}$ delimits a range of explored guesser complexity.[1] Once initialized, generations start running. A spot is randomly chosen for evaluation of the guesser that it contains. The likelihood of picking up a 1-guesser is twice the likelihood of picking up a 2-guesser, three times the likelihood

[1] Here, $n^{\max}$ marks a range of guesser complexity. Guessers with higher complexity are not allowed by design. This was used in Ref. [38], but it becomes irrelevant in the experiments of the current manuscript, as we will see.

of picking up a 3-guesser, and so on. In this way, in average, guessers are offered a chance to guess a same number of bits.

The guesser picked up for evaluation (a 5-guesser in Fig. 9) is presented with a newly generated m-environment. This environment is created randomly and on the spot, such that the ecosystem as a whole is evaluated over a subset of the m-environment ensemble ($\hat{E}^m \subset E^m$, as described above). The selected guesser first comes up with the best guess given the specific m-environment that has been generated. This implies two important things: (i) We are again looking at limitations set up by the guesser complexity, as we decide to evaluate the best inference possible given the size, n, of the guesser (which determines its computational capabilities). (ii) All n-guessers are equivalent to each other given the environment. This is so because we wish to find differences, hopefully, that are only due to the computational specifications of guessers and environments. Note, however, that two different n-guessers with a same n evaluated at different times will be confronted with different m-environments—thus their best guesses would differ (because they are determined by the environment). Note also that if we would evaluate a same guesser twice on a same specific m-environment, while it would end up with a same internal model Γ, this might generate two different behaviors (W) because the later also depend on the position of the environment onto which the guesser is dropped. Note that 2-guessers in Fig. 9(a) show different W.

Guessers accumulate reward over time ($\rho(t)$) as described in the main text. If this reward is kept positive, then the guesser survives. If this reward doubles a "metabolic load" ($\rho > 2n\rho_0$), then the guesser gets replicated [Fig. 9(b)] and produces a new guesser with the same n. This new guesser occupies a random position of the ecosystem. This random position might be empty or occupied by another guesser. In this case, the older guesser is substituted by the new one. If $\rho(t) < 0$, then the guesser dies [Fig. 9(c)]. Its spot in the ecosystem is occupied by a 0-guesser (i.e., an empty space). When computing the average complexity of an environment (i.e., the average n of the guessers contained), empty spaces are scored as a 0-guesser. This is how ecosystem complexity in Fig. 6 of the main text drops to 0.

Besides m, experiments with simple ecosystems have fixed values of other model parameters such as α and ρ_0. Together, they determine what guesser complexity survives and which dies off. For example, very stringent environments have large α and they offer very little reward per guessed bit (as compared to the cost of attempting a guess). In these cases, more complex guessers (which make more correct guesses per attempt) might be favored. Furthermore, in the current paper we have introduced a static parasite to such environments (Sec. III C of the main text). This also affects what guessers survive, as shown in Fig. 6.

More interesting experiments are introduced by the coevolution of hosts and parasites—and they bring about the most alluring results in the paper concerning Red Queen dynamics. To implement these experiments we introduce what we called a *shadow* ecosystem, which is nothing but the possibility that each guesser in a main ecosystem (which we refer as hosts) might be infected each by a single parasite. Shadow ecosystems are depicted as a box behind the main ecosystem in Fig. 10. In this version of the experiments, host guessers are picked up for evaluation just as before (in Fig. 10, a 9-guesser is selected). If there is a guesser in the same position of the shadow ecosystem, then it is selected for evaluation as well. (In the same figure, there is a 3-guesser parasite.) Evaluation of the host proceeds as usual. Evaluation of the guesser proceeds as explained in the Methods section of the main text, depending on whether we are dealing with a genotype or phenotype parasite. The parasites in Fig. 10 affect genotypes. The reward acquired by the parasite is subtracted from the host's accumulated reward.

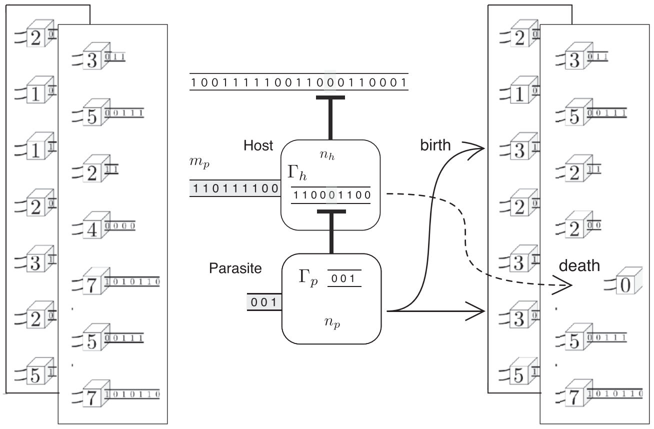

FIG. 10. Ecoevolutionary dynamics. Host-parasite dynamics in ecosystems are implemented through a shadow ecosystem such that parasites are picked for evaluation along with their hosts.

If the host's reward becomes negative ($\rho(t) < 0$), then it dies and it is removed from the experiment, leaving an empty spot (0-guesser) behind. If the host was infected by some parasite (i.e., if there was a parasite in the same position of the shadow ecosystem), then it dies as well (again, leaving a 0-guesser behind in the shadow ecosystem). If the host doubles its replication threshold ($\rho(t) > 2n\rho_0$), then it gets replicated. In doing so, it might displace an earlier host guesser as explained above. If the host carries a parasite, then it gets replicated as well. The minimal reward needed to establish the daughter parasite ($\bar{n}\bar{\rho}_0$) is subtracted from the daughter host. It might seem that hosts are attached to their parasites forever, but a parasite might die because its reward becomes negative after an unsuccessful evaluation ($\bar{\rho}(t) < 0$). Also, parasites might replicate independently of their host if they accrue a reward:

$$\bar{\rho}(t) > 2\bar{n}\bar{\rho}_0. \tag{A1}$$

In this case, the daughter parasite will occupy a new, random spot in the shadow ecosystem. By doing so, it might displace an earlier parasite—thus changing the parasite that previously infected some other host. Parasite replication is resolved before host death, so that the daughter of the parasite of a dying host (as in Fig. 10) lives on.

The model is limited in that guessers must be smaller than their environment. To ensure this, if a daughter parasite was relocated to a spot in which a host is less complex than the incoming parasite, then the host is removed from the experiment altogether and the parasite is promoted to the main ecosystem. We understand that this is what happens if a parasite is relocated to an empty spot, where a 0-guesser dwelt. Our experiments indicate that this mechanism is seldom used, since hosts and parasites sustain a gap of complexity that prevents parasites from becoming more complex than hosts.

The final element of this experiment is the evolutionary dynamics. Besides the mode of replication just described, both hosts and parasites might mutate. Mutation is enabled only if a guesser accrues enough reward to satisfy the metabolic load of a guesser with just a unit longer [i.e., $(n+1)\rho_0$ for hosts, $(\bar{n}+1)\bar{\rho}_0$ for parasites]. That is, mutation is possible if

$$\rho(t) > (2n+1)\rho_0$$

for hosts and if

$$\bar{\rho}(t) > (2\bar{n}+1)\bar{\rho}_0$$

for parasites. If this happens, then mutation ensues with probability p_μ. If a mutation takes place, then it results in a guesser of less complexity ($n \rightarrow n-1$ for hosts, $\bar{n} \rightarrow \bar{n}-1$ for parasites) or in a guesser of more complexity ($n \rightarrow n-1$ for hosts, $\bar{n} \rightarrow \bar{n}-1$ for parasites) with equal chance.

[1] J. J. Hopfield, Physics, computation, and why biology looks so different, J. Theor. Biol. **171**, 53 (1994).

[2] E. Szathmáry and J. Maynard-Smith, From replicators to reproducers: The first major transitions leading to life, J. Theor. Biol. **187**, 555 (1997).

[3] G. F. Joyce, Molecular evolution: Booting up life, Nature (London) **420**, 278 (2002).

[4] S. I. Walker and P. C. Davies, The algorithmic origins of life, J. R. Soc., Interface **10**, 20120869 (2013).

[5] E. Smith and H. J. Morowitz, *The Origin and Nature of Life on Earth: The Emergence of the Fourth Geosphere* (Cambridge University Press, Cambridge, UK, 2016).

[6] S. J. Gould, *Wonderful Life: The Burgess Shale and the Nature of History* (WW Norton & Company, New York, NY, 1990).

[7] E. Jablonka and M. J. Lamb, The evolution of information in the major transitions, J. Theor. Biol. **239**, 236 (2006).

[8] F. Jacob, *Of Flies, Mice, and Men* (Harvard University Press, Cambridge, MA, 1998).

[9] C. R. Woese, On the evolution of cells, Proc. Natl. Acad. Sci. USA **99**, 8742 (2002).

[10] C. Darwin, *On the Origin of Species*, edited by J. Murray (1859).

[11] D. C. Dennet, *Darwin's Dangerous Idea: Evolution and the Meanings of Life* (Simon & Schuster, New York, NY, 1995).

[12] I. G. Johnston, K. Dingle, S. F. Greenbury, C. Q. Camargo, J. P. Doye, S. E. Ahnert, and A. A. Louis, Symmetry and simplicity spontaneously emerge from the algorithmic nature of evolution, Proc. Natl. Acad. Sci. USA **119**, e2113883119 (2022).

[13] R. Fisher, *The Genetical Theory of Natural Selection* (The Clarendon Press, Oxford, UK, 1930).

[14] S. J. Gould, *The Structure of Evolutionary Theory* (Harvard University Press, Cambridge, MA, 2002).

[15] S. A. Frank, Natural selection maximizes Fisher information, J. Evol. Biol. **22**, 231 (2009).

[16] B. McMullin, John von Neumann and the evolutionary growth of complexity: Looking backward, looking forward, Artific. Life **6**, 347 (2000).

[17] M. A. Bedau, J. S. McCaskill, N. H. Packard, S. Rasmussen, C. Adami, D. G. Green, T. Ikegami, K. Kaneko, and T. S. Ray, Open problems in artificial life, Artific. Life **6**, 363 (2000).

[18] K. Ruiz-Mirazo, J. Umerez, and A. Moreno, Enabling conditions for "open-ended evolution," Biol. Philos. **23**, 67 (2008).

[19] T. Day, Computability, Gödel's incompleteness theorem, and an inherent limit on the predictability of evolution, J. R. Soc., Interface **9**, 624 (2011).

[20] T. Taylor, M. Bedau, A. Channon, D. Ackley, W. Banzhaf, G. Beslon, E. Dolson, T. Froese, S. Hickinbotham, T. Ikegami, and B. McMullin, Open-ended evolution: Perspectives from the OEE workshop in York, Artific. Life **22**, 408 (2016).

[21] H. P. de Vladar, M. Santos, and E. Szathmáry, Grand views of evolution, Trends Ecol. Evol. **32**, 324 (2017).

[22] B. Corominas-Murtra, L. F. Seoane, and R. Solé, Zipf's law, unbounded complexity and open-ended evolution, J. R. Soc., Interface **15**, 20180395 (2018).

[23] S. J. Gould, *Full House* (Harvard University Press, Cambridge, MA, 2011).

[24] R. Solé, M. Moses, and S. Forrest, Liquid brains, solid brains, Philos. Trans. R. Soc. London B **374**, 20190040 (2019).

[25] J. Iranzo, P. Puigbo, A. E. Lobkovsky, Y. I. Wolf, and E. V. Koonin, Inevitability of genetic parasites, Genome Biol. Evol. **8**, 2856 (2016).

[26] E. Koonin, Y. Wolf, and M. Katsnelson, Inevitability of the emergence and persistence of genetic parasites caused by evolutionary instability of parasite-free states, Biol. Direct **12**, 31 (2017).

[27] R. Solé and S. F. Elena, *Viruses as Complex Adaptive Systems* (Princeton University Press, Princeton, NJ, 2018).

[28] T. J. Little, The evolutionary significance of parasitism: Do parasite-driven genetic dynamics occur *ex silico*? J. Evol. Biol. **15**, 1 (2002).

[29] T. S. Ray, An approach to the synthesis of life, in *Artificial Life II*, edited by C. Langton, C. Taylor, and D. Farmer (Addison-Wesley, Redwood City, 1991), 880, pp. 371–408.

[30] R. E. Lenski, C. Ofria, R. T. Pennock, and C. Adami, The evolutionary origin of complex features, Nature (London) **423**, 139 (2003).

[31] C. Adami, Digital genetics: Unravelling the genetic basis of evolution, Nat. Rev. Genet. **7**, 109 (2006).

[32] W. D. Hamilton, R. Axelrod, and R. Tanese, Sexual reproduction as an adaptation to resist parasites: A review, Proc. Natl. Acad. Sci. USA **87**, 3566 (1990).

[33] W. D. Hillis, Co-evolving parasites improve simulated evolution as an optimization procedure, Physica D **42**, 228 (1990).

[34] T. Ikegami and K. Kaneko, Computer symbiosis—Emergence of symbiotic behavior through evolution, Physica D **42**, 235 (1990).

[35] L. Zaman, J. R. Meyer, S. Devangam, D. M. Bryson, R. E. Lenski, and C. Ofria, Coevolution drives the emergence of complex traits and promotes evolvability, PLoS Biol. **12**, e1002023 (2014).

[36] S. Brenner, Biological computation, Limits Reduct. Biol. **213**, 106 (1998).

[37] P. Nurse, Life, logic and information, Nature (London) **454**, 424 (2008).

[38] L. F. Seoane and R. V. Solé, Information theory, predictability, and the emergence of complex life, R. Soc. Open Sci. **5**, 172221 (2018).

[39] J. E. Hopcroft, Turing machines, Sci. Am. **250**, 86 (1984).

[40] C. H. Bennett, The thermodynamics of computation—A review, Int. J. Theor. Phys. **21**, 905 (1982).

[41] C. Adami, What is complexity? BioEssays **24**, 1085 (2002).

[42] A. N. Kolmogorov, 1965 Three approaches to the quantitative definition of information, Probl. Inf. Transm. **1**, 1 (2002).

[43] T. M. Cover and J. A. Thomas, *Elements of Information Theory* (Wiley, New York, NY, 1999).

[44] J. Ziv and A. Lempel, A universal algorithm for sequential data compression, IEEE Trans. Inf. Theory **23**, 337 (1977).

[45] T. Welch, A technique for high-performance data compression, Computer **17**, 8 (1984).

[46] See Supplemental Material at http://link.aps.org/supplemental/10.1103/PhysRevE.108.044407 for additional figures further illustrating the model.

[47] R. J. Solomonoff, A preliminary report on a general theory of inductive inference, Tech. Rept. ZTB-138, Zator Company, Cambridge, MA (1960).

[48] G. J. Chaitin, Algorithmic information theory, IBM J. Res. Dev. **21**, 350 (1977).

[49] G. J. Chaitin, *Information, Randomness & Incompleteness: Papers on Algorithmic Information Theory*, Vol. 8 (World Scientific, Singapore, 1990).

[50] R. Solé, Revisiting Leigh van Valen's "A new evolutionary law" (1973), Biol. Theory **17**, 120 (2022).

[51] N. A. Heim, J. L. Payne, S. Finnegan *et al.*, Hierarchical complexity and the size limits of life, Proc. R. Soc. London B **284**, 20171039 (2017).

[52] S. J. Hickinbotham, S. Stepney, and P. Hogeweg, Nothing in evolution makes sense except in the light of parasitism: Evolution of complex replication strategies, R. Soc. Open Sci. **8**, 210441 (2021).

[53] L. Sagan, On the origin of mitosing cells, J. Theor. Biol. **14**, 225-IN6 (1967).

[54] L. Bonen and W. F. Doolittle, On the prokaryotic nature of red algal chloroplasts, Proc. Natl. Acad. Sci. USA **72**, 2310 (1975).

[55] R. Dawkins, C. Leelayuwat, S. Gaudieri *et al.*, Genomics of the major histocompatibility complex: Haplotypes, duplication, retroviruses, and disease, Immunol. Rev. **167**, 275 (1999).

[56] S. Mi Jr, X. Lee, X. P. Li, G. M. Veldman *et al.*, Syncytin is a captive retroviral envelope protein involved in human placental morphogenesis, Nature (London) **403**, 785 (2000).

[57] L. P. Villarreal and V. R. DeFilippis, A hypothesis for DNA viruses as the origin of eukaryotic replication proteins, J. Virol. **74**, 7079 (2000).

[58] P. J. L. Livingstone Bell, Viral eukaryogenesis: Was the ancestor of the nucleus a complex DNA virus? J. Mol. Evol. **53**, 251 (2001).

[59] J. F. Hughes and J. M. Coffin, Evidence for genomic rearrangements mediated by human endogenous retroviruses during primate evolution, Nat. Genet. **29**, 487 (2001).

[60] J. B. Whitfield, Estimating the age of the polydnavirus/braconid wasp symbiosis, Proc. Natl. Acad. Sci. USA **99**, 7508 (2002).

[61] P. Forterre, The origin of DNA genomes and DNA replication proteins, Curr. Opin. Microbiol. **5**, 525 (2002).

[62] S. Blaise, N. de Parseval, L. Bénit, and T. Heidmann, Genomewide screening for fusogenic human endogenous retrovirus envelopes identifies syncytin 2, a gene conserved on primate evolution, Proc. Natl. Acad. Sci. USA **100**, 13013 (2003).

[63] D. Marti, C. Grossniklaus-Bürgin, S. Wyder, T. Wyler, and B. Lanzrein, Ovary development and polydnavirus morphogenesis in the parasitic wasp *Chelonus inanitus*. I. Ovary morphogenesis, amplification of viral DNA and ecdysteroid titres, J. Gen. Virol. **84**, 1141 (2003).

[64] T. Wyler and B. Lanzrein, Ovary development and polydnavirus morphogenesis in the parasitic wasp *Chelonus inanitus*. II. Ultrastructural analysis of calyx cell development, virion formation and release, J. Gen. Virol. **84**, 1151 (2003).

[65] E. Espagne, C. Dupuy, E. Huguet *et al.*, Genome sequence of a polydnavirus: Insights into symbiotic virus evolution, Science **306**, 286 (2004).

[66] P. Kiepiela, A. J. Leslie, I. Honeyborne, D. Ramduth, C. Thobakgale, S. Chetty, P. Rathnavalu, C. Moore, K. J. Pfafferott, L. Hilton, and P. Zimbwa, Dominant influence of HLA-B in mediating the potential coevolution of HIV and HLA, Nature (London) **432**, 769 (2004).

[67] J. M. Claverie, Viruses take center stage in cellular evolution, Genome Biol. **7**, 110 (2006).

[68] J. M. Claverie, H. Ogata, S. Audic, C. Abergel, K. Suhre, and P. E. Fournier, Mimivirus and the emerging concept of "giant" virus, Virus Res. **117**, 133 (2006).

[69] P. J. L. Bell, The viral eukaryogenesis hypothesis, Ann. N.Y. Acad. Sci. **1178**, 91 (2009).

[70] P. Forterre and D. Prangishvili, The great billion-year war between ribosome-and capsid-encoding organisms (cells and viruses) as the major source of evolutionary novelties, Ann. N.Y. Acad. Sci. **1178**, 65 (2009).

[71] N. Lane, *Life Ascending: The Ten Great Inventions of Evolution* (WW Norton & Company, New York, NY, 2010).

[72] E. V. Koonin, *The Logic of Chance: The Nature and Origin of Biological Evolution* (FT Press, Upper Saddle River, NJ, 2011).

[73] G. Witzany ed., *Viruses: Essential Agents of Life* (Springer Science & Business Media, Cham, 2012).

[74] V. Sundaram, Y. Cheng, Z. Ma *et al.*, Widespread contribution of transposable elements to the innovation of gene regulatory networks, Genome Res. **24**, 1963 (2014).

[75] N. Lane, *The Vital Question: Energy, Evolution, and the Origins of Complex Life* (WW Norton & Company, New York, NY, 2015).

[76] E. B. Chuong, N. C. Elde, and C. Feschotte, Regulatory evolution of innate immunity through co-option of endogenous retroviruses, Science **351**, 1083 (2016).

[77] E. B. Chuong, N. C. Elde, and C. Feschotte, Regulatory activities of transposable elements: From conflicts to benefits, Nat. Rev. Genet. **18**, 71 (2017).

[78] G. Cornelis *et al.*, An endogenous retroviral envelope syncytin and its cognate receptor identified in the viviparous placental Mabuya lizard, Proc. Natl. Acad. Sci. USA **114**, E10991 (2017).

[79] V. Chaikeeratisak, K. Nguyen, K. Khanna *et al.*, Assembly of a nucleus-like structure during viral replication in bacteria, Science **355**, 194 (2017).

[80] L. P. Villarreal and F. Ryan, Viruses in the origin of life and its subsequent diversification, in *Handbook of Astrobiology*, edited by V. M. Kolb (CRC Press, Boca Raton, FL, 2018).

[81] R. Mizuuchi, T. Furubayashi, and N. Ichihashi, Evolutionary transition from a single RNA replicator to a multiple replicator network, Nat. Commun. **13**, 1460 (2022).

[82] R. Landauer, Irreversibility and heat generation in the computing process, IBM J. Res. Dev. **5**, 183 (1961).

[83] J. T. Bonner, *The Evolution of Complexity by Means of Natural Selection* (Princeton University Press, Princeton, NJ, 1988).

[84] M. Eigen, Natural selection: A phase transition? Biophys. Chem. **85**, 101 (2000).

[85] P. Davies, *The Demon in the Machine: How Hidden Webs of Information Are Solving the Mystery of Life* (University of Chicago Press, Chicago, IL, 2019).

[86] D. R. Amor, R. Montañez, S. Duran-Nebreda, and R. Solé, Spatial dynamics of synthetic microbial mutualists and their parasites, PLoS Comput. Biol. **13**, e1005689 (2017).

[87] M. C. Boerlijst and P. Hogeweg, Spiral wave structure in prebiotic evolution: Hypercycles stable against parasites, Physica D **48**, 17 (1991).

[88] J. Sardanyés and R. V. Solé, Spatio-temporal dynamics in simple asymmetric hypercycles under weak parasitic coupling, Physica D **231**, 116 (2007).

[89] A. L. Barabasi, V. W. Freeh, H. Jeong, and J. B. Brockman, Parasitic computing, Nature (London) **412**, 894 (2001).

[90] M. Lewis and D. Baker, *Flash Boys* (WW Norton, New York, NY, 2014).

[91] M. Del Giudice, Invisible designers: Brain evolution through the lens of parasite manipulation, Q. Rev. Biol. **94**, 249 (2019).

[92] I. Goodfellow, J. Pouget-Abadie, and M. Mirza *et al.*, Generative adversarial nets, in *Advances in Neural Information Processing Systems* (MIT Press, Cambridge, MA, 2014), pp. 2672–2680.

[93] A. Radford, L. Metz, and S. Chintala, Unsupervised representation learning with deep convolutional generative adversarial networks, arXiv:1511.06434.