Optimization of Time-Ordered Processes in the Finite and Asymptotic Regimes

Mirjam Weilenmann[1,2,*] Costantino Budroni[3] and Miguel Navascués[2]

[1]*Département de Physique Appliquée, Université de Genève, Genève, Switzerland*

[2]*Institute for Quantum Optics and Quantum Information—IQOQI Vienna, Austrian Academy of Sciences, Boltzmanngasse 3, Vienna 1090, Austria*

[3]*Department of Physics "E. Fermi", University of Pisa, Largo B. Pontecorvo 3, Pisa 56127, Italy*

Many problems in quantum information theory can be formulated as optimizations over the sequential outcomes of dynamical systems subject to unpredictable external influences. Such problems include many-body entanglement detection through adaptive measurements, computing the maximum average score of a preparation game over a continuous set of target states, and limiting the behavior of a (quantum) finite-state automaton. In this work, we introduce tractable relaxations of this class of optimization problems. To illustrate their performance, we use them to: (a) compute the probability that a finite-state automaton outputs a given sequence of bits; (b) develop a new many-body entanglement-detection protocol; and (c) let the computer *invent* an adaptive protocol for magic state detection. As we further show, the maximum score of a sequential problem in the limit of infinitely many time steps is in general incomputable. Nonetheless, we provide general heuristics to bound this quantity and show that they provide useful estimates in relevant scenarios.

I. INTRODUCTION

In quantum physics, we are usually concerned with manipulating quantum systems, whether we interact with them to cause them to reach (or remain in) a specific quantum state or measure them in order to observe or certify some of their properties. Finding new, better, ways to perform tasks with time-ordered operations lies at the heart of current research in quantum information science. Quantum computing relies on a sequential application of gates to a quantum system and communication protocols aided by quantum systems depend on performing operations on these systems and exchanging information in a specific order. Sequences of operations are also crucial in less evident situations; e.g., specific sequences of pulses are applied in order to read out a superconducting qubit [1].

Traditionally, the problem of finding new ways (or sequences of operations) to complete a specific task has been dependent on the ingenuity of scientists to come up with new protocols. More recently, this type of problem is also tackled via machine-learning techniques, which require training a neural network to generate the right sequence of operations. However, a general method to optimize over sequential strategies or even a criterion to decide whether one such sequential protocol is optimal for a certain task is lacking.

Part of the difficulty in solving these problems lies in the fact that when operations are performed sequentially, protocols can be made adaptive, i.e., future operations may depend on the result of previous ones. Adaptiveness can lead to great advantages; e.g., it minimizes statistical errors in entanglement certification [2] and quantum state detection [3]. However, it also increases the variety of protocols that must be considered for a specific task and thus the complexity of optimizing over them. Indeed, while optimization techniques are well established in the context of single-shot or independently repeated procedures (e.g., in applications of nonlocality [4,5] or for witnessing entanglement [6]), developing them for sequential protocols remains a challenge.

In this work, we make progress on this problem. Specifically, we introduce a general technique to analyze and optimize a class of time-ordered processes that we call *sequential models*. These are protocols in N rounds, where in each round an interaction occurs and changes the state of a system of interest possibly depending on unknown

*Corresponding author: mirjam.weilenmann@unige.ch

or uncontrolled variables. The progress toward a certain goal is quantified by a reward that is generated at each round of the protocol and is the object of our optimization procedure.

Our first result is a technique that allows us to upper bound the maximum total reward that a sequential model can generate in N rounds through a complete hierarchy of increasingly difficult convex-optimization problems. To illustrate the performance of the hierarchy, we apply it to upper bound the type-I error of an adaptive protocol for many-body entanglement detection [2,7]. The technique generates seemingly tight bounds for large system sizes (of order $N \approx 100$).

The same technique also allows us to deepen the study of temporal correlations generated by *finite-state automata* [8,9]. These models appear in several problems in quantum foundations and quantum information theory, such as classical simulations of quantum contextuality [10–13], quantum simulations of classical stochastic processes [14–16], purity certification [17], dimension witnesses [18–20], quantum advantages in the design of time-keeping devices [21–24], and classical simulation of quantum computation [25]; for more details, see a recent review on temporal correlations [26]. Specifically, we focus on the problem of optimal clocks, namely, how good a clock can we construct, given access to a classical automaton with d internal states? Using our tools, we solve this problem for low values of d, by providing upper bounds on the maximum probability that the automaton outputs the "one-tick sequence" $000 \cdots 01$ investigated in Refs. [23,24]. These bounds match the best known lower bounds up to numerical precision.

In addition to finite sequences of operations, we are also often interested in letting a protocol or process run indefinitely, i.e., we are interested in characterizing its asymptotic behavior as $N \to \infty$. This is, e.g., important for systems that are left to evolve for a long time or in cases in which we aim to probe large systems, where this limit is a good approximation.

To our knowledge, not much is known about time-ordered processes in the limit of infinitely many rounds (except for results on asymptotic rates in hypothesis testing and when taking specific limits [27,28]). In this paper, we show that there is a fundamental reason for this: there are sequential models for which no algorithm can approximate their asymptotic behavior. This follows from undecidability results related to finite-state automata; specifically, a construction from Refs. [29,30]. Nevertheless, we develop a heuristic method for computing rigorous bounds on the asymptotic behavior. We further find that in the applications we consider, these bounds are close to the expected asymptotic behavior and to the lower bounds we can compute, thus substantiating the usefulness of our method. More specifically, we use the asymptotic method to bound the type-I error of the many-body entanglement-detection

protocol mentioned above in the limit of infinitely many particles. The result is a bound that is, at most, at a distance of 4×10^{-4} from the actual figure. Our asymptotic method also bounds the probability that a two-state automaton outputs the one-tick sequence in the limit of many time steps to $O(10^{-2})$.

Finally, in some circumstances it becomes necessary not to analyze but to find sequential models with a good performance. That is, given a number of parameters that we can control—the policy—which might affect the evolution of the system as well as the reward, we pose the problem of deciding which policy maximizes the total reward after N rounds. We propose to tackle this problem via projected-gradient methods and show how to cast the computation of the gradient of our upper bounds as a tractable convex-optimization problem. Using this approach, we let the computer discover a two-state six-round preparation game to detect one-qubit magic states.

Our paper is structured as follows. In Sec. II, we present three problems in quantum information theory that we will use to illustrate our methods throughout. In Sec. III, we introduce the abstract notion of sequential models and show how to optimize them, before applying the optimization method in Sec. IV. In Sec. V, we prove that the asymptotic behavior of sequential models cannot be approximated in general. We also introduce a heuristic to tackle asymptotic problems, which we apply in Sec. VI. In Sec. VII, we pose the problem of policy optimization and show how to compute the gradient of the upper bounds derived in Sec. III, over the parameters of the policy. We use gradient methods to solve a policy optimization problem in Sec. VIII. Finally, we present our conclusions in Sec. IX.

II. PROBLEMS WITH TIME-ORDERED OPERATIONS

A variety of different problems involving quantum systems are of a sequential nature. In the following, we introduce three settings that are made up of time-ordered operations and that all pose challenging open problems to the quantum information community.

A. Temporal correlations

A *finite-state automaton* (FSA) is a mathematical structure that models an autonomous computational device with bounded memory. More formally [8,9], a d-state automaton is a device described, at every time, by an internal state $\sigma \in \Sigma$, with $|\Sigma| = d$. At regular intervals, rounds or time steps, the automaton updates its internal state and generates an outcome $b \in B$, depending on both its prior internal state σ and the state $y \in Y$ of its input port. Such a double state-and-output transition is governed by the transition matrix $P(\sigma', b | \sigma, y)$ of the automaton, which indicates the probability that the automaton transitions to the internal

state σ' and outputs b, given that its input and prior states were, respectively, y and σ. Overall, the probability of outputs $\mathbf{b} = (b_1, \ldots, b_n)$ given inputs $\mathbf{y} = (y_1, \ldots, y_n)$ can be computed as

$$p(\mathbf{b}|\mathbf{y}) = \sum_{\sigma_0, \sigma_1, \ldots, \sigma_n} p(\sigma_0) P(\sigma_1, b_1|\sigma_0, y_1) P(\sigma_2, b_2|\sigma_1, y_2) \ldots$$

$$\times \, P(\sigma_n, b_n|\sigma_{n-1}, y_n). \tag{1}$$

Expressions such as Eq. (1), i.e., generated by a classical FSA, appear in several problems related to classical simulations of temporal quantum correlations, such as, e.g., classical simulations of quantum contextuality [10–12], quantum simulations of classical stochastic processes [14–16], classical simulation of quantum computation [25], and quantum advantages in the design of time-keeping devices [21–24].

We are interested in scenarios in which the performance of a d-state automaton after n time steps is evaluated by yet another automaton or, more generally, by a time-dependent process with bounded memory. We wish to bound said performance over all automata with d states.

This class of problems includes, e.g., computing the maximum probability that a d-state automaton generates the so-called "one-tick sequence" $00\ldots01$ (the name comes from its interpretation as a clock signal "1" for a tick of the clock). This optimization problem is central in the discussion of the optimal classical and quantum models able to generate a time signal, with the consequent quantum advantages in the design of physical clocks. Unfortunately, there are no general optimization methods to upper bound the performance of a classical model and thus demonstrate a quantum advantage. The problem of optimal quantum clocks has been extensively investigated both from a theoretical (quantum information and quantum thermodynamics) perspective [21–24,31–36], as well as from an experimental one [37].

Consider thus an FSA A, with states denoted by $\sigma \in \Sigma$, with $|\Sigma| = d$, inputs $y \in Y$ and outputs $b \in B$. We assume that the automaton is described by the (unknown) transition matrix $P(\sigma_{k+1}, b_k|\sigma_k, y_k)$. Let Q be a known time-dependent process, with an internal state denoted by t and inputs (outputs) denoted by z (c). The (known) transition matrix of this process at time steps $k = 2, \ldots, N$ is $Q_k(t_k, c_k|t_{k-1}, z_{k-1})$. Each state t_k of the process is associated with a reward $\alpha_k(t_k)$. At time step 1, the state and output t_1, c_1 are generated by the distribution $Q_1(t_1, c_1)$. Now, let us couple Q with A, in the following way (see also Fig. 1 for an illustration): after Q generates t_1, c_1, we input c_1 in the automaton A, i.e., we let $y_1 = c_1$. The automaton then transitions through P to a state σ_1, outputting b_1. This output is further input into the process Q (by letting $z_1 = b_1$), which in turn generates t_2, c_2 through Q_2, and so on. At time step N, we consider the sum of all rewards,
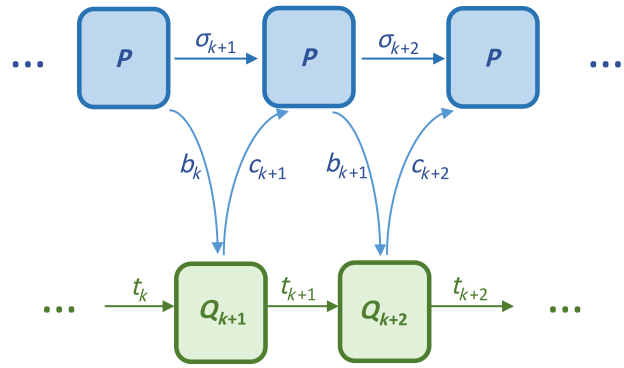

FIG. 1.   An automaton (blue) coupled to a process Q (green). The process may, in general, differ from round to round.

$\sum_{k=1}^{N} \alpha_k(t_k)$. Our goal is to maximize the expectation value of the total reward over all d-state automata A. Note that by convexity, the best strategy is for the automaton to be initialized in a specific state $\sigma_0 = \bar{\sigma}_0$ rather than a distribution thereof [38]. Hence, fixing the initial state, the optimal automaton is fully characterized in terms of the transition matrix P that maximizes the expected reward.

This problem can be mathematically rephrased in a way that will prove useful later. Namely, the coupled systems, A and Q, can be regarded as a single dynamical system the internal state s_k of which at time step k corresponds to the probability distribution of the triple (t_k, c_k, σ_k), i.e., $s_k := p_k(t_k, c_k, \sigma_k)$. The evolution of this system from one time step to the next follows the equation of motion,

$$p_{k+1}(t_{k+1}, c_{k+1}, \sigma_{k+1})$$

$$= \sum_{c_k, t_k, \sigma_k, b_k} p_k(t_k, c_k, \sigma_k) P(\sigma_{k+1}, b_k|\sigma_k, c_k)$$

$$\times \, Q_{k+1}(t_{k+1}, c_{k+1}|t_k, b_k)$$

$$=: f_k(s_k, \lambda). \tag{2}$$

Note that the evolution is given here by a polynomial f_k of the state s_k and an unknown evolution parameter $\lambda := (P(\sigma', c|\sigma, z) : z \in Z, c \in C, \sigma \in \Sigma)$. In this formulation of the problem, the reward r_k at time step k is given by

$$r_k(s_k) := \sum_{t_k, c_k, \sigma_k} p_k(t_k, c_k, \sigma_k) \alpha(t_k). \tag{3}$$

Note that the reward does not depend on λ here (except via the p_k): such a dependency could, however, be introduced by using different reward functions $r_k(s_k, \lambda)$. In this picture, our task is to maximize the total reward over all possible evolution parameters λ.

B. Entanglement certification of many-body systems

Given an N-partite quantum system, we are often faced with the problem of determining whether its state is entangled. In principle, this problem can be solved by finding an appropriate entanglement witness and estimating its value on the state at hand. However, in addition to the problem that finding such a witness theoretically is NP-hard [39], its estimation may involve conducting joint measurements on many subsystems of the N-partite state, a feat that may not be experimentally possible. Replacing such joint measurements by estimates based on local measurement statistics requires, in general, $\mathrm{Exp}[N]$ state preparations, thus rendering the corresponding protocols infeasible for moderately sized N (as is the case when the protocol demands full quantum state tomography).

One way to avoid these problems, i.e., perform single-system measurements and reduce the number of state preparations, is to carry out an adaptive protocol [2]. Such a scheme proceeds as follows. We sequentially measure the particles that constitute the N-particle ensemble, making future measurements depend on previous ones as well as on previous outcomes. Once we conduct the last measurement, we make a guess on whether the underlying state was entangled or not.

To choose how to measure each system and to make our final guess, we use a dynamical system Q. This process has internal states $t_k \in T$ and its transition matrix at time $k = 1, \ldots, N - 1$ is of the form $Q_k(t_{k+1}, y_{k+1}|t_k, b_k)$, where $b_k \in B$ denotes the measurement outcome at time k and y_{k+1} labels the measurements to be conducted on the $(k + 1)$th particle. For each $y \in Y_k$, there exists a positive-operator-valued measure (POVM) $\{M_{b|y}^k\}_b$. The final guess is generated through the function $Q_N(t_{N+1}|t_N, b_N)$, with $t_{N+1} \in \{\text{"entangled," "separable"}\}$. The initial state of the process and the measurement setting y_1 for the first particle are generated by the distribution $Q_0(t_1, y_1)$. This is illustrated in Fig. 2.

If the target state ρ the entanglement of which we wish to detect experimentally admits an efficient matrix product operator (MPO) decomposition [40], then one can efficiently compute the probability that the process outputs the result "separable" (see Fig. 3). This probability is usually called a *type-II* error or a *false negative*.

It remains to compute the *type-I* error, or *false positive*, of the entanglement-detection protocol encoded in Q: that would correspond to the maximum probability that the process outputs "entangled" when the input is a fully separable quantum state. In computing this maximum, we can assume, by convexity, that the player has prepared a pure product quantum state. That is, at each time step k, we measure some pure quantum state ρ_k. Despite this simplification, computing the type-I error of the scheme requires maximizing a multilinear function with a size-$O(N)$ input. As N grows, this becomes a more difficult problem.
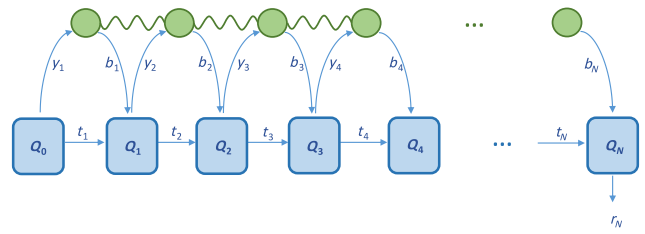

FIG. 2. The sequential model for entanglement detection. The particles of an N-party state (green) are measured one by one. The kth measurement y_k depends on the internal state of a finite-state automaton and the outcome b_k affects the transition of the automaton according to Q_k.

Like the problem of computing the maximum performance over all d-state automata, maximizing the type-I error over the set of separable states can be phrased as an optimization over the sequential outputs of a deterministic dynamical system. In this case, the internal state s_k of the dynamical system at time step k would correspond to the distribution P_k of t_k, y_k. The equation of motion of the system is

$$P_{k+1}(t_{k+1}, y_{k+1}) = \sum_{t_k, y_k, b_k} P_k(t_k, y_k) \mathrm{tr}(M_{b_k|y_k}^k \rho_k)$$
$$\times Q_k(t_{k+1}, y_{k+1}|t_k, b_k), \quad (4)$$

where the quantum state ρ_k can be interpreted as an uncontrolled external variable h influencing the evolution of the system. At time step N, the system outputs the
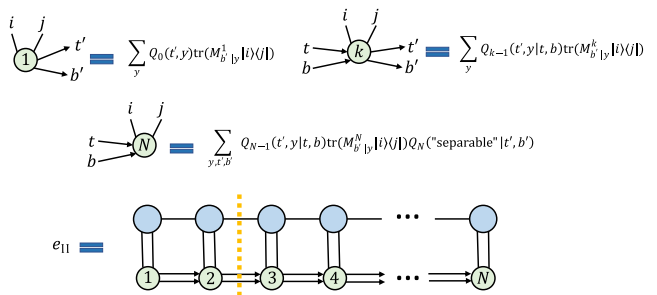

FIG. 3. Efficient computation of the type-II error. The blue circles represent the tensors making up our MPO target state, with two vertical legs denoting the bra and ket indices of each particle and two horizontal legs of dimension D—the bond dimension—to account for the correlations of the target state. Defining the green tensors in the way indicated in the figure, their contraction with the blue tensors equals the type-II error of the corresponding entanglement-detection protocol. This contraction can be computed by multiplying matrices of dimension $|T||B|D$, as the dashed orange line indicates.

deterministic outcome

$$r_N(s_N) = \sum_{t_N, y_N, b_N} P_N(t_N, y_N) \mathrm{tr}(M^N_{b_N|y_N} \rho_N)$$

$$\times Q_N(t_{N+1}|t_N, b_N) \delta_{t_{N+1}, \text{"entangled"}}. \quad (5)$$

C. Independent identically distributed (IID) preparation strategies in quantum preparation games

The notion of preparation games aims to capture the structure of general (possibly adaptive) protocols, where quantum systems are probed sequentially [2]. More precisely, a *quantum preparation game* is an N-round task involving a player and a referee. In each round, the player prepares a quantum state, which is then probed by the referee. In round k, before carrying out his measurement, the referee's current knowledge of the source used by the player is encoded in a variable $t_k \in T_k$, with $|T_k| < \infty$, called the *game configuration*. This variable depends nontrivially on the past history of measurements and measurement results: in each round, it guides the referee in deciding which measurement to perform next and changes depending on its outcome. This double role of the game configuration can be encoded in the POVM used by the referee for that round. That is, assuming that the game configuration before the measurement is $t \in T_k$, the probability that the new game configuration is $t' \in T_{k+1}$ is given by $\mathrm{tr}(\rho_k M^k_{t'|t})$, where ρ_k is the player's kth-state preparation; and $\{M^k_{t'|t} : t' \in T_{k+1}\}$, the POVM implemented by the referee when the game configuration is t. At the end of the preparation game, a score $g(t_{N+1})$, where t_{N+1} denotes the final game configuration, is generated by the referee according to some scoring system $g(t)$ for $t \in T_{N+1}$.

In some circumstances—e.g., in an entanglement-detection protocol where the player tries to (honestly) prepare a specific entangled state—it makes sense to consider a player that follows an independent identically distributed (IID) strategy, meaning that the player produces the same state ρ in each round.

Consider thus the problem of computing the maximum game score, for a fixed scoring system, achievable with preparation strategies of the form $\rho^{\otimes N}$, with $\rho \in C$, where C is some set of states. In Ref. [2], it has been shown how to compute the maximum score of a preparation game under IID strategies when the set C of feasible preparations has finite cardinality. In this work, we consider the case in which C is continuous.

This problem can be modeled through a dynamical system the internal state s_k of which at time k corresponds to $P_k(t_k)$, the distribution of game configurations at the beginning of round k. The equation of motion of the system is

$$P_{k+1}(t') = \sum_{t \in T_k} P_k(t) \mathrm{tr}(\rho M^k_{t'|t}), \quad (6)$$

where the unknown evolution parameters $\lambda := \rho \in C$ correspond to the player's preparation. The dynamical system outputs a reward at time N, namely,

$$r_N(s_N, \lambda) = \sum_{t' \in T_{N+1}} g(t') \sum_{t \in T_k} P_N(t) \mathrm{tr}(\rho M^N_{t'|t}). \quad (7)$$

Our goal is thus to maximize the reward of the system over all possible values of the evolution parameter λ, i.e., over all $\rho \in C$.

As an aside, we note that in some circumstances it becomes necessary to compute the maximum average game score achievable by an *adversarial player*, who has access to the current game configuration and can thus adjust their preparation in each round accordingly. Curiously enough, this problem is easier than optimizing over IID strategies, and, in fact, a general solution is presented in Ref. [2]. Notwithstanding, we show below that the computation of the maximum game score of a preparation game under adaptive strategies can also be regarded as a particular class of the sequential problems considered in this paper.

III. SEQUENTIAL MODELS AND THEIR OPTIMIZATION

The three problems described above are—even though from a physical point of view rather different—structurally very similar. Indeed, they are all examples of the type of problem that we introduce more abstractly in the following. Consider a scenario in which the state of a dynamical system is fully described at time step k by a variable $s_k \in S_k$. Between time steps k and $k+1$, the system transitions from s_k to another state s_{k+1} depending on s_k, a number of uncontrolled variables $h_k \in H_k$ and a number of unknown evolution parameters $\lambda \in \Lambda$. The equation of motion that describes this transition is

$$s_{k+1} = f_k(s_k, h_k, \lambda), \quad (8)$$

where S_k, H_k $\forall k$, and Λ are sets of parameter values.

Our goal is to optimize over this type of system, according to some figure of merit that is given by the problem at hand. We model this by assuming that, in each time step, the system emits a reward (or penalty) $r_k(s_k, h_k, \lambda)$. Note that this reward may be the zero function for certain k, so that this includes problems where only the final outcome matters as a special case. We shall refer to such a system from now on as a *sequential model* (see Fig. 4).

Our goal is to estimate the maximum total reward $\nu^\star$ over all possible values of $\lambda, h_1, \ldots, h_N$. That is, we wish
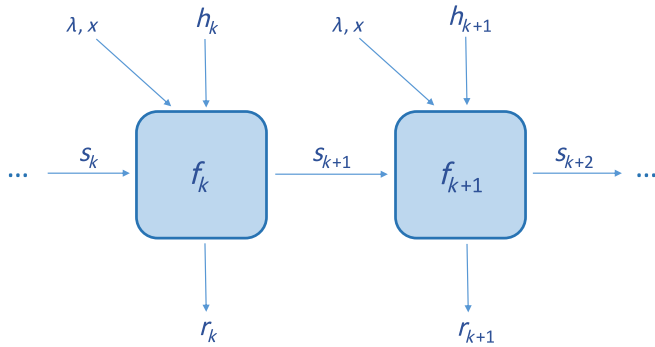
FIG. 4. A pictorial representation of a sequential model. Note that the equation of motion itself may capture several interactions or be composed of several steps (e.g., the measurement of a quantum system followed by the update of a finite-state automaton).

to solve the problem

$$
\begin{aligned}
\nu^\star := \max_{\lambda, h} \quad & \sum_{k=1}^{N} r_k(s_k, h_k, \lambda) \\
\text{such that} \quad & \lambda \in \Lambda, \ h \in \times_{k=1}^{N} H_k \\
& s_{k+1} = f_k(s_k, h_k, \lambda) \quad \forall k.
\end{aligned}
\tag{9}
$$

We generally assume that the initial state s_1 is known; otherwise, we can absorb it into the definition of λ in the sense that we add one time step that generates the initial state. In the absence of λ, the above is mathematically equivalent to a deterministic Markov decision problem, if we regard h as an action [41].

A. Reformulation of the optimization over sequential models

To compute the optimal value $\nu^\star$, we consider an approach based on dynamical programming methods [41]. It consists in defining *value functions* $V_k(s_k, \lambda)$, which represent the maximum reward achievable between time steps k and N over all possible values of $h_k, \ldots, h_N$, starting from the state s_k, and under the assumption that the evolution parameters take the value λ. Approximating these functions with polynomials and invoking known characterizations of positive polynomials on a semialgebraic set, we manage to derive monotone converging sequences of tractable upper bounds on $\nu^\star$. We remark that similar ideas have been explored in the context of continuous-time optimal-control problems, which can be formulated as an infinite-dimensional linear program, for which semidefinite-program (SDP) relaxations based on polynomial optimization techniques exist; see Ref. [42] and references therein.

In terms of the reward functions, value functions are given by

$$
\begin{aligned}
V_N(s_N, \lambda) &= \max_{h_N \in H_N} r_N(s_N, h_N, \lambda), \\
V_k(s_k, \lambda) &= \max_{h_k \in H_k} r_k(s_k, h_k, \lambda) + V_{k+1}(f_k(s_k, h_k, \lambda), \lambda),
\end{aligned}
\tag{10}
$$

and where we recover $\nu^\star$ in the final step as

$$
\nu^\star = \max_{\lambda \in \Lambda} V_1(s_1, \lambda).
\tag{11}
$$

The problem of finding $\nu^\star$ can thus be reduced to

$$
\begin{aligned}
\min_{V_1, \ldots, V_N, \nu} \quad & \nu \\
\text{such that} \quad & V_N(s_N, \lambda) \geq r_N(s_N, h, \lambda) \quad \forall h \in H_N, \ s_N \in S_N, \ \lambda \in \Lambda \\
& V_k(s_k, \lambda) \geq r_k(s_k, h, \lambda) + V_{k+1}(f_k(s_k, h, \lambda), \lambda) \quad \forall h \in H_k, \ s_k \in S_k, \ \lambda \in \Lambda \\
& \nu \geq V_1(s_1, \lambda) \quad \forall \lambda \in \Lambda.
\end{aligned}
\tag{12}
$$

Indeed, on one hand, any feasible point of the problem given in Eq. (12) provides an upper bound on $\nu^\star$. On the other hand, the $V_1, \ldots, V_N, \nu^\star$, as defined by Eqs. (10) and (11), are also a feasible point of Eq. (12). Hence the solution of the problem given in Eq. (12) is $\nu^\star$.

Unfortunately, unless their domain is finite, there is no general method for optimization over arbitrary functions. What is feasible is to solve the problem given in Eq. (12)

under the assumption that $\{V_k(s_k, \lambda)\}_k$ belong to a class of functions $\mathcal{F}$ described by finitely many parameters. Since, in general, the optimal functions $V_k(s_k, \lambda)$ might not belong to $\mathcal{F}$, the result $\bar{\nu}$ of such an optimization is an upper bound on the actual solution $\nu^\star$ of the problem. The class $\mathcal{F}$ of polynomial functions is very handy, as it is closed under composition and dense in the set of continuous functions with respect to the uniform norm. This is the class we are working on in this paper.

For the rest of the paper, we thus assume that, for all $k \in \{1, 2, \ldots, N\}$, $f_k(s_k, h_k, \lambda)$, $r_k(s_k, h_k, \lambda)$ are polynomials on s_k, h_k, λ and that the sets S, H_k, and Λ are bounded sets in some metric space that can be described by a finite number of polynomial inequalities. Note that under these assumptions, the value functions are continuous on s_k and λ.

Call, then, $\nu^n \geq \nu^\star$ the result of Eq. (12) under the constraint that $\{V_k(s_k, \lambda)\}_k$ are polynomials of a given degree n. Since any continuous function defined on a bounded set can be arbitrarily well approximated by polynomials, it follows that $\lim_{n \to \infty} \nu^n = \nu^\star$.

There is the added difficulty that solving Eq. (12) entails enforcing positivity constraints of the form

$$p(x) \geq 0, \quad \forall x \in X, \tag{13}$$

where $p(x)$ is a polynomial on the vector variable $x \in \mathbb{R}^m$ and X is a bounded region of $\mathbb{R}^m$ defined by a finite number of polynomial inequalities (i.e., a basic semialgebraic set). Fortunately, there exist several complete (infinite) hierarchies of sufficient criteria to prove the positivity of a polynomial on a semialgebraic set [43–46]. In algebraic geometry, any such hierarchy of criteria is called a *positivstellensatz*.

Two prominent positivstellensätze are Schmüdgen's [45] and Putinar's [46]. Both hierarchies have the peculiarity that, for a fixed index k, the set of all polynomials satisfying the kth criterion is representable through an SDP. The application of Schmüdgen's and Putinar's positivstellensätze for polynomial optimization thus leads to two complete hierarchies of SDP relaxations. The SDP relaxation based on Putinar's positivstellensatz is known as the Lasserre-Parrilo hierarchy [47,48]. The reader can find a description of both hierarchies in Sec. I of the Supplemental Material [49]. While the hierarchy based on Schmüdgen's positivstellensatz is more computationally demanding than the Lasserre-Parrilo hierarchy, it converges faster. In our numerical calculations below, we use a hybrid of the two.

The use of the kth (sufficient) positivity test of a given positivstellensatz while optimizing over polynomial value functions of degree n results in an upper bound ν_k^n on ν^n, with $\lim_{k \to \infty} \nu_k^n = \nu^n$. For our purposes, the take-home message is that $\nu_k^n \geq \nu^\star$, for all k, n and $\lim_{n \to \infty} \lim_{k \to \infty} \nu_k^n = \nu^\star$.

We finish this section by noting that there is nothing fundamental about the use of polynomials to approximate value functions. In fact, there exist other complete families of functions that one might utilize to solve the problem given in Eq. (12). A promising such class is the set of *signomials*, or linear combinations of exponentials, for which there also exists a number of positivsellensätze in the literature [50,51]. In this case, the corresponding sets of positive polynomials are not SDP representable but

are nonetheless tractable convex sets. As such, they are suitable for optimization.

B. Quantum preparation games as sequential models

Sequential models capture the structure of various types of time-ordered processes, including the three introduced in Sec. II. Here, we argue that sequential models are actually rather general: optimizations over adversarial strategies in quantum preparation games can be modeled as sequential problems too. In fact, their resolution by means of value functions allows one to rederive previous results on the topic [2].

Let $\mathcal{S}$ be a set of quantum states and consider a preparation game where the player is allowed to prepare, at each round k, any state $\rho(t_k) \in \mathcal{S}$ depending on the current game configuration t_k. This preparation game can be interpreted as a sequential model with $\Lambda = \emptyset$, $s_k = p_k(t_k)$, $h_k = (\rho^k(t) : t \in T_k)$ and an equation of motion

$$p_{k+1}(t') = \sum_{t \in T_k} p_k(t) \text{tr}(\rho^k(t) M_{t'|t}^k). \tag{14}$$

We consider the problem of maximizing the average game score (or reward)

$$r_{N+1}(p_{N+1}) = \sum_{t \in T_{N+1}} p_{N+1}(t) g(t), \tag{15}$$

where g is the score function of the game [2].

In principle, we could formulate this problem as Eq. (12). Note that $V_{N+1} = r_{N+1}$ is linear in p_{N+1}. This motivates us to consider an ansatz of linear value functions $V_k(p_k)$ for this problem. That is, we aim to solve Eq. (12) under the assumption that

$$V_k(p_k) = \sum_{t \in T_k} \mu_k(t) p_k(t), \tag{16}$$

for some vector $(\mu_k(t) : t \in T_k)$. The condition $V_k(p_k) \geq V_{k+1}(p_{k+1})$ then translates to

$$\sum_{t \in T_k} \mu_k(t) p_k(t)$$

$$\geq \max_{\{\rho^k(t)\}_t} \sum_{t' \in T_{k+1}} \mu_{k+1}(t') \sum_{t \in T_k} p_k(t) \text{tr}(\rho^k(t) M_{t'|t}^k)$$

$$= \sum_{t \in T_k} p_k(t) \max_{\rho^k(t)} \text{tr}\left(\rho^k(t) \sum_{t' \in T_{k+1}} V_{k+1}(t') M_{t'|t}^k\right). \tag{17}$$

Setting $p_k(t) = \delta_{t,j}$, for $j \in T_k$, we arrive at the recursion relations

$$\mu_{N+1}(t) = g(t),$$

$$\mu_k(t) = \max_{\rho^k(t)} \mathrm{tr}\left(\rho^k(t) \sum_{t' \in T_{k+1}} \mu_{k+1}(t') M_{t'|t}^k \right), \quad (18)$$

which allow us to compute $V_1(t_0)$. This quantity is, in principle, an upper bound on the actual solution of Eq. (12), because we have enforced the extra constraint that the value functions are linear. The upper bound is, however, tight, as can be seen by computing the average score of the preparation game under the preparation strategy given by the maximizers $\bar{\rho}^k(t)$ of Eq. (18). In fact, all the above is but a complicated way to arrive at the recursion relations provided in Ref. [2] to compute the maximum score of a preparation game.

IV. APPLICATION TO OPTIMIZING TIME-ORDERED PROCESSES

All the sequential problems considered in this work could, in principle, be solved (or at least approximated) through the straightforward application of the Lasserre-Parrilo hierarchy [47,48]. However, the degree of the polynomials in such problems grows linearly with the number of steps N, thus making the corresponding SDP intractable for more than a few iterations. Our reformulation of the sequential optimization problem as in Eq. (12), on the other hand, allows us to circumvent this issue and optimize sequential models for much higher values of N.

A. Optimizations over finite-state automata: Probability bounds

In this section, we investigate the maximum probability with which a classical FSA can generate a given output sequence. This problem has been extensively investigated for input-output sequences [12], as well as sequences with only inputs [23,24]. In particular, we consider the so-called one-tick sequence, $00\ldots01$, introduced in Ref. [23] in connection with the problem of optimal clocks. For a sequence of total length L, we use the compact notation $0^{L-1}1$. Here, we use the general techniques developed in Sec. III to tackle this problem. Note that this is an instance of the type of problem introduced in Sec. II A.

Consider a d-state automaton without inputs and with binary outputs, i.e., $Y = \emptyset$, $B = \{0, 1\}$. We wish to compute the maximum probability that the automaton outputs the one-tick sequence $0^{L-1}1$, denoted $P_{\max}^d(L)$. For this purpose, let us construct a simple process Q that, coupled to the automaton, produces an expected reward that equals $P(0^{L-1}1)$. Q has two internal states, i.e., $t_k \in \{0, 1\}$ and no outputs ($C = \emptyset$). Its initial state is $t_1 = 1$, and its transition

matrices are as follows:

$$Q_k(t_{k+1}|t_k, b_k) = \delta_{t_{k+1}, t_k(1-b_k)}, \quad \text{for } k = 1, \ldots, L-1,$$

$$Q_L(t_{L+1}|t_L, b_L) = \delta_{t_{L+1}, t_L b_L}. \quad (19)$$

For time steps $k = 1, \ldots, L-1$, the process Q therefore keeps being in state 1 as long as the automaton outputs 0's; if the automaton outputs any 1, the state of the process changes to 0 and stays there until the end of the game. At time step L, process Q receives the output b_L. If $b_L = 0$ or $t_L = 0$, then $t_{L+1} = 0$; otherwise, $t_{L+1} = 1$. The probability that the automaton produces the sequence $0^{L-1}1$ thus corresponds to the probability that $t_{L+1} = 1$. This corresponds to a reward function

$$\alpha_k(t_k) = 0, \quad \text{for } k = 1, \ldots, L-1,$$

$$\alpha_{L+1}(t_{L+1}) = \delta_{t_{L+1}, 1}. \quad (20)$$

Using the techniques developed in Sec. III, we map this problem to a polynomial optimization problem. Due to the choice of reward (together with the process Q), we see that the only terms from the equation of motion given in Eq. (2) that contribute to the final reward are the ones with the correct b_k. This allows a further simplification of the model.

First, let us redefine the internal states of the sequential model to be the vector

$$s_k = (p_k(t_k = 1, \sigma_{k+1}))_\sigma, \quad (21)$$

described by d parameters s_k^σ, with constraints $s_k^\sigma \geq 0$ for $\sigma = 0, \ldots, d-1$, and $1 - \sum_\sigma s_k^\sigma \geq 0$. Next, we redefine the unknown evolution parameter λ to have dimension d^2, through $\lambda := (P(\sigma', b = 0|\sigma) : \sigma, \sigma' = 0, 1)$, with constraints

$$P(\sigma', b = 0|\sigma) \geq 0, \quad \text{for all } \sigma, \sigma' = 0, \ldots, d-1,$$

$$1 - \sum_{\sigma'} P(\sigma', b = 0|\sigma) \geq 0, \quad \text{for } \sigma = 0, \ldots, d-1. \quad (22)$$

For $k = 1, \ldots, L-1$, the equation of motion of this sequential model is

$$p_{k+1}(t_k = 1, \sigma_{k+1}) = \sum_{\sigma_k} p_k(t_{k-1} = 1, \sigma_k) P(\sigma_{k+1}, 0|\sigma_k), \quad (23)$$

while the reward function reads

$$r_k = 0, \quad \text{for } k = 1, \ldots, L-1;$$

$$r_L(s_L, \lambda) = \sum_\sigma p_L(t_L = 1, \sigma)\left(1 - \sum_{\sigma'} P(\sigma', b = 1|\sigma)\right). \quad (24)$$
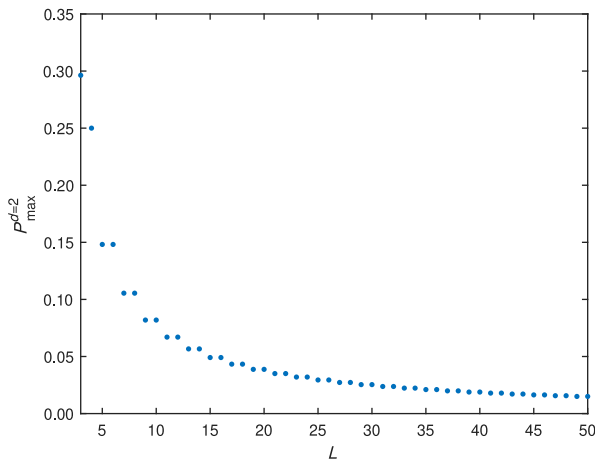
FIG. 5. Optimization over two-state automata. The upper bounds (red) follow a pattern, where from $L = 5$ on pairs of consecutive values coincide. Our bounds furthermore match the explicit automata found in Ref. [23] up to numerical precision, thus providing a tight bound and proving the optimality of the models found in Refs. [23,24]. The optimization has been performed with MOSEK [52] and CVXPY [53]. Each solution provided by the solver has been tested to verify that linear and positivity constraints are satisfied up to numerical precision.

Now, we apply an SDP relaxation à la Lasserre [47] to obtain an upper bound on the solution of the sequential problem for the cases of $d = 2$ and $d = 3$. In the latter case, no nontrivial analytical upper bound has previously been
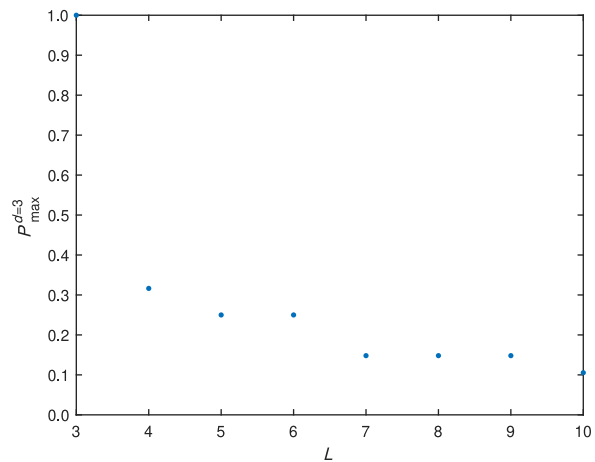

FIG. 6. Optimization over three-state automata. Our method allows us also to optimize over three-state automata. Again, we recover, up to numerical precision, the bound computed with the explicit model found in Refs. [23,24], proving that our bound is tight and that the model previously found was optimal. The optimization has been performed with SCS [54] and CVXPY [53]. Each solution provided by the solver has been tested to verify that linear and positivity constraints are satisfied up to numerical precision.

known [23,24]. For details regarding the exact implementation, we refer to Sec. IIA of the Supplemental Material [49]. In Figs. 5 and 6, we display the upper bounds computed with this method. We find that, in every case, the obtained upper bound matches the output probability of the automaton proposed in Refs. [23,24], up to numerical precision. This implies that the upper bounds computed are all tight. The optimal models for these sequences, except in the case $L = d + 1$, correspond to the so-called cyclic model [23]. The model consists of deterministic transitions from one state to the other, and the probability of emitting 1 is nonzero only in the last state. As a consequence, the bound depends only on $\lfloor L/d \rfloor$, corresponding to the 2- and 3-periodic plateaus observed in Figs. 5 and 6.

In order to compare the performance of our algorithm with those of the previously known method, namely, the Lasserre-Parrilo hierarchy [47,48], we also perform a numerical optimization of our problem with the latter method. While our method has been able to reach $L = 50$ for $d = 2$ and $L = 10$ for $d = 3$, the Lasserre-Parrilo hierarchy could not go beyond the case $L = 7$ for $d = 2$. This boost in performance was to be expected, as the degree of the polynomials in the Lasserre-Parrilo hierarchy grows linearly with the number of steps L, whereas in our method it remains constant. For more details, see Secs. I and II of the Supplemental Material [49].

In addition to the numerical values for $P_{\text{max}}^{\text{d}}$, we can also extract the value functions from our optimizations and use those to prove our bounds analytically. After deriving those value functions, we can further use them to find optimal automata for the problem at hand.

For extracting value functions, let us note first that from the numerical solutions, the coefficients of the polynomials V_k can be directly extracted, as these are optimization variables of the problem. However, the value functions may not be unique and extracting polynomials with suitable (rational or integer) coefficients from the numerics is more challenging. Ways to further simplify these value functions are elaborated in Sec. IIB of the Supplemental Material [49].

The method for optimizing sequential models employed here allows us to derive upper bounds in terms of value functions, without treating the actual variables of the model—in this case, the $\{p_k\}_k$ and P—explicitly [55]. *A priori*, the solution hence does not tell us what the values of the optimal dynamical systems achieving the bounds may be, even if the bounds are tight. However, this can be remedied by first extracting the value functions from the optimization and then running a second optimization, this time fixing the value functions and optimizing for the variables of the sequential model, here $\{p_k\}_k$ and P. More precisely, if we have found an optimal upper bound $v^\star$, then all value function inequalities have to be tight and there is an optimal dynamical system that achieves the bound. Now if we can find a system that satisfies all

inequalities with equality, then this will automatically be optimal. To find such a model, we can thus consider the optimization problem

$$\max_{s_k, \lambda, h_k} V_1(s_1, \lambda)$$

subject to $V_N(s_N, \lambda) = r_N(s_N, h_N, \lambda)$

$$V_k(s_k, \lambda) = r_k(s_k, h_k, \lambda) + V_{k+1}(f_k(s_k, h_k, \lambda), \lambda)$$

$$h_k \in H_k \quad \forall k, \ s_k \in S_k \ \forall k, \ \lambda \in \Lambda.$$

$$(25)$$

This is again a polynomial optimization problem that can be solved with an SDP relaxation. In the following examples, we use the SOLVEMOMENT functionality of YALMIP [56] at level 4 and the solver MOSEK [52] to solve this [57]. We find that in these cases, the problem is feasible and we extract optimal automata.

(a) In the case $d = 2$ and $L = 3$ above, we obtain

$$V_1(s_1, \lambda) = (P(0,0|0)q_1(0)$$
$$+ P(0,0|1)q_1(1))P(b = 1|0)$$
$$+ (P(1,0|0)q_1(0)$$
$$+ P(1,0|1)q_1(1))P(b = 1|1), \quad (26)$$

where $q_k(\sigma) := p_k(1, \sigma)$. Note that in this case, this is the natural optimization problem that one would also formulate directly to solve the problem at hand; however, for higher L, the decomposition into value functions differs. Now, using that the final reward is

$$r_2 = P(b = 1|0)q_2(0) + P(b = 1|1)q_2(1) \quad (27)$$

and the state-transition rules that translate to

$$q_2(0) = P(0,0|0)q_1(0) + P(0,0|1)q_1(1), \quad (28)$$
$$q_2(1) = P(1,0|0)q_1(0) + P(1,0|1)q_1(1), \quad (29)$$

we immediately see that $V_1(s_1, \lambda) - r_2(s_2(s_1), \lambda) = 0$. Furthermore, we can derive upper bounds on $V_1(s_1(s_0), \lambda)$, where we have from the state-transition rules $q_1(0) = P(0,0|0)$ and $q_1(1) = P(1,0|0)$ (as the starting state is $q_0(1) = 1, q_0(2) = 0$) that

$$V_1(s_1(s_0), \lambda) = (P(1,0|1)^2$$
$$+ P(2,0|1)P(1,0|2))P(b = 1|1)$$
$$+ (P(1,0|1)P(2,0|1)$$
$$+ P(2,0|1)P(2,0|2))P(b = 1|2).$$
$$(30)$$

This gives a polynomial upper bound on the solution that indeed is upper bounded by 0.29630, as

a numerical maximization of the value function over the P and q_k confirms. The optimization also shows that this maximum is achieved by an automaton with $P(0,0|0) = P(1,0|1) = \frac{1}{3}$, $P(1,0|0) = \frac{2}{3}$, $P(0,0|1) = 0$, thus confirming this value as the optimal tight upper bound [58]. This also coincides with the optimal automaton proposed in Ref. [23].

(b) For $d = 2$ and $L = 4$, we obtain the two value functions [59]

$$W_2(s_2, \lambda) = V_1(s_2, \lambda) \quad (31)$$

$$W_1(s_1, \lambda) = 0.25(1 + q_1(0)(q_1(0) - P(0,0|0))$$
$$+ q_1(1)(q_1(1) - P(1,0|0))) \quad (32)$$

Now, we can see that 0.25 is an upper bound for $W_1(s_1, \lambda)$, as the state-update rules and $q_0(0) = 1$ and $q_0(1) = 0$ in this case imply that $q_1(0) = P(0,0|0)$ and $q_1(1) = P(1,0|0)$. As a last relation [60], we have to check that

$$W_1(s_1, \lambda) - W_2(s_2(s_1), \lambda) \geq 0. \quad (33)$$

A numerical optimization confirms this polynomial inequality. Requiring that $W_1(s_1, \lambda) = W_2(s_2(s_1), \lambda)$ while maximizing $W_1(s_1, \lambda)$ under the constraints on s_1 and s_2 gives an optimal solution for the automaton of $P(1,0|0) = 1$, $P(0,0|0) = 0$ and $P(0,0|1) = \frac{1}{2}$, $P(1,0|1) = 0$ [61]. This again coincides with the automaton from [23] for $L = 4$.

Note that there are usually many different value functions that could solve the problem. Extracting value functions with few terms and integer or fractional coefficients from the optimization problem, however, requires some tuning (see Sec. IIB of the Supplemental Material [49]). The process becomes more cumbersome as we increase the number of variables. We thus do not provide value functions for larger L here. We remark that extracting valid value functions from the optimization is, nevertheless, always possible.

B. Entanglement detection in many-body systems: The automaton-guided GHZ game

To illustrate the performance of our SDP relaxations to solve entanglement certification problems in many-body systems, we focus our attention on the N-qubit Greenberger-Horne-Zeilinger (GHZ) state

$$|\Psi_{\text{GHZ}}\rangle = \frac{1}{\sqrt{2}} \left(|0\rangle^{\otimes N} + |1\rangle^{\otimes N}\right) \quad (34)$$

of size up to $N = 35$. This is a timely application, as GHZ states of sizes up to 20 qubits can already be prepared

[62–64]. The main goal in devising such protocols is to obtain a procedure that requires as few state preparations as possible, for reliably certifying entanglement. In this section, we propose such a one-shot protocol and demonstrate that following this procedure indeed needs fewer repetitions n than a family of protocols based on shadow tomography [65].

The main feature of our method is that we work in a hypothesis-testing setting. This means that we are also able to prove bounds on the performance of the protocol for the worst-case separable state (corresponding to the type-I error), which is difficult to compute and generally not provided when estimating fidelities. Our method has the additional advantage of relying only on single-qubit quantum operations (as compared, e.g., to the 18-qubit example from Ref. [62] that uses two-qubit gates not just in the preparation but also in the disentangling phase) and it can also deal with protocols that are adaptive (see Sec. IV C), (as compared to determining the expectation of an operator, e.g., the parity [63,64], by fixed measurements or compared to evaluating Bell inequalities [66]). This latter aspect is what generally allows us to reduce the number of repetitions of our protocols.

It is worth remarking that, in *any* one-shot entanglement-detection protocol that aims to detect the pure GHZ state, the type-I and type-II errors e_I, e_{II} will satisfy the relation

$$e_I + e_{II} \geq \frac{1}{2}. \tag{35}$$

To see why, call M the N-qubit effective POVM element (depending on the protocol, the corresponding POVM will be global, local operations and classical communication (LOCC), or one-way LOCC) associated with declaring the system "entangled" and let σ be the separable state

$$\sigma = \frac{1}{2}(|0\rangle\langle 0|^{\otimes N} + |1\rangle\langle 1|^{\otimes N}). \tag{36}$$

Then, we have that

$$1 - e_{II} - e_I(\sigma) = \text{tr}\{M(|\Psi_{GHZ}\rangle\langle\Psi_{GHZ}| - \sigma)\}$$

$$\leq \frac{1}{2} \||\Psi_{GHZ}\rangle\langle\Psi_{GHZ}| - \sigma\|_1$$

$$= \frac{1}{2}, \tag{37}$$

where $e_I(\sigma)$ is the probability of declaring σ "entangled." Since $e_I \geq e_I(\sigma)$, we arrive at Eq. (35).

The protocol that we propose in the following saturates the relation in Eq. (35) in the limit $N \to \infty$. In this sense, despite solely relying on one-qubit measurements, our protocol is asymptotically optimal.

Let us, then, describe our entanglement-detection protocol. The protocol is inspired by the GHZ game, previously

considered in Refs. [67,68], where here the role of the memory is taken by a classical FSA. We thus call this protocol the *automaton-guided GHZ game*. Specifically, picture a protocol for many-body entanglement verification where, for each of the first $N - 1$ particles, one of the following two measurements is performed with probability $\frac{1}{2}$:

$$M_1 = \{M_{1|1} = |+\rangle\langle+|, M_{2|1} = |-\rangle\langle-|\},$$
$$M_2 = \{M_{1|2} = |+i\rangle\langle+i|, M_{2|2} = |-i\rangle\langle-i|\}, \tag{38}$$

where $|\pm\rangle = \frac{1}{\sqrt{2}}(|0\rangle \pm |1\rangle)$ and $|\pm i\rangle = \frac{1}{\sqrt{2}}(|0\rangle \pm i|1\rangle)$. This process is used to update the state of a four-state automaton that is initialized in state $t_1 = 1$ as displayed in Fig. 7. For the final particle, we choose measurement 1 for states $t_N = \pm 1$ of the automaton and measurement 2 for $t_N = \pm i$. The verification of an entangled state is considered successful if the outcome, b_N, of this additional measurement is $b_N = 1$ for $t_N \in \{1, i\}$ or $b_N = 2$ for $t_N \in \{-1, -i\}$, in which case we update t_{N+1} to "entangled"; otherwise "separable."

This protocol succeeds with probability 1 for a GHZ state of any dimension. Indeed, think of an k-partite GHZ-like state of the form

$$\frac{1}{\sqrt{2}}\left(|0\rangle^{\otimes k} + \alpha|1\rangle^{\otimes k}\right), \tag{39}$$

where $\alpha \in \{1, -1, i, -i\}$. Suppose that we measure one of the qubit states in the basis $\{|+\rangle, |-\rangle\}$ ($\{|i\rangle, |-i\rangle\}$) and obtain the result $|\pm\rangle$ ($|\pm i\rangle$). Then, the final postselected
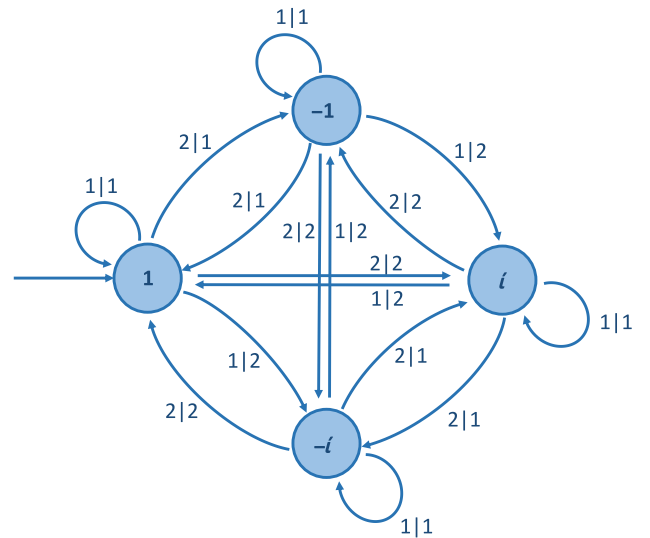

FIG. 7. The memory update of the four-state automaton in the GHZ game. The transitions between states depend deterministically on b and y, which are used to label the edges as $b|y$. In each round of the associated sequential model, this updates the automaton state $t \in \{1, -1, i, -i\}$.
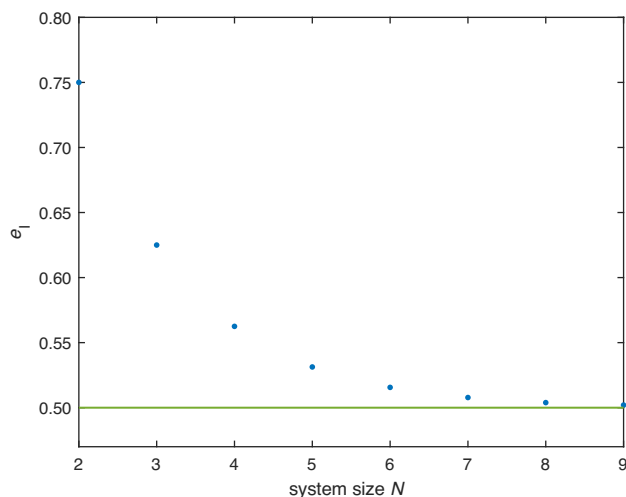
FIG. 8. The upper bounds for the winning probability of the best separable states in the GHZ game. We observe that the winning probability (blue) drops quickly toward 0.5 (green line). The bounds are displayed up to $N = 9$ for aesthetic reasons. In fact, we have checked that we can derive bounds up to $N = 35$ with this method. With our asymptotic hierarchy (see Sec. VI B), we further confirm an upper bound on the asymptotic value of 0.5004 as $N \to \infty$.

state is an $(k-1)$-partite state of the form given in Eq. (39) but this time with phase $\pm \alpha$ ($\mp i\alpha$).

Due to its transition matrix, the internal state of the automaton in Fig. 7 keeps track of the phase of the GHZ-like state, as its qubits get sequentially measured, i.e., $t = \alpha$. The reward function simply verifies that the state of the last qubit corresponds to $\frac{1}{\sqrt{2}}(|0\rangle + \alpha|1\rangle)$.

The procedure introduced in Sec. III allows us to derive upper bounds on the maximum probability of success that is achievable with separable states, i.e., the worst-case type-I error e_I. We display our bounds for different system sizes n in Fig. 8. Note that the size of the optimization problem allows us to reach round numbers as high as $N = 35$. As the reader can appreciate, e_I seems to converge to $\frac{1}{2}$ in the limit $N \to \infty$, thus saturating the inequality in Eq. (35). Details on the implementation (including precision issues and how to resolve them in this case) are presented in Sec. IIC of the Supplemental Material [49].

We further compare the detection performance of the automaton-guided GHZ game with a family of entanglement-detection protocols based on shadow tomography [65], in the following called *shadow protocols*.

There exist different protocols for shadow tomography, depending on the type of measurements conducted on the qubits. Since for the automaton-guided GHZ game we only allow the measurement of each qubit individually, we opt for shadow protocols involving random measurements in the three Pauli bases.

Our family of shadow protocols for entanglement detection thus works as follows:

(1) A random measurement in one of the Pauli bases is carried out on each qubit and both the result and the basis are recorded. This procedure is repeated for n experimental rounds (or state preparations).

(2) The resulting classical shadow is then used to produce an estimator f for the underlying fidelity of the quantum state with respect to the GHZ state.

(3) If f is greater than a threshold $0 \le \theta \le 1$, then the protocol outputs the result "entangled"; otherwise, it outputs "separable."

There is thus one shadow protocol for each value of θ.

The computation of the type-II error $e_{II}(\theta)$ for this family of protocols is straightforward—not so the computation of $e_I(\theta)$, due to the need to optimize the probability that the protocol outputs "entangled" over the set of separable input states. Because of this, we replace the maximization over all separable states by a maximization over the three separable states $|0\rangle^{\otimes N}$, $|+\rangle^{\otimes N}$, and $|+i\rangle^{\otimes N}$. The resulting quantity $\tilde{e}_I(\theta)$ is therefore a lower bound on the actual value of $e_I(\theta)$, i.e., $e_I(\theta) \ge \tilde{e}_I(\theta)$. Since our aim is to show that the automaton-guided GHZ game outperforms the shadow protocols, underestimating the error in the latter is sufficient, as it provides us with a lower bound on the advantages that we observe for the automaton-guided GHZ game.

For different values of θ, we have estimated the type-I and type-II errors $\tilde{e}_I(\theta), e_{II}(\theta)$ of the shadow protocols for the 5-partite GHZ state in $n = 50$ experimental rounds [69]. To compare these numbers against the performance of the one-round automaton-guided GHZ game, we have extended the latter to n experimental repetitions in the following way:

(1) Run the automaton-guided GHZ game for n experimental rounds, noting down the number of times n_e that the protocol outputs "entangled."

(2) If n_e/n exceeds the threshold $0 \le \phi \le 1$, declare the state entangled; otherwise, separable.

Pairs of values (e_I, e_{II}) for both families of protocols are plotted in Fig. 9. Note that the curve for the automaton-guided GHZ games lies completely below that of the shadow protocols. In fact, for some values of the threshold ϕ, the automaton-guided GHZ game exhibits values with $e_I \approx e_{II} = 0$. This was to be expected. Indeed, let $\bar{e}_I \approx 0.53$ be the type-I error of the one-shot GHZ game. Then, for $\phi = 1$, the entanglement of the GHZ state is still detected with probability 1 (and so $e_{II} = 0$). Furthermore, a separable state can only be mislabeled as entangled with probability upper bounded by $e_I = \bar{e}_I^{50} \approx 0$.

One might have expected that shadow-based protocols have the upper hand for large values of the system size N. Nonetheless, from Fig. 8, it is clear that the $(e_I(\phi), e_{II}(\phi))$ curve of the automaton-guided protocol will improve with
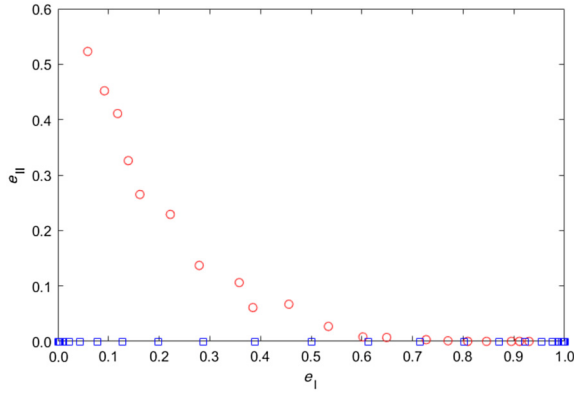
FIG. 9. Type-I errors versus type-II errors. The blue squares denote the automaton method; the red circles, the shadow protocols. For some values of the threshold ϕ, the automaton method is essentially perfect.

increasing N. This is not the case for shadow-based GHZ fidelity estimation (see Ref. [65]). Thus, for larger N, the advantage of our automaton-guided GHZ game will be even larger. Finally, we remark that we have compared our automaton-guided GHZ game with the shadow-based GHZ fidelity estimation of Ref. [65]. However, a better estimate on the GHZ fidelity can be obtained by selecting a different sampling rule, e.g., Paulis from the stabilizer group, following the idea of Ref. [70].

C. Entanglement detection in many-body systems: The bit-sequence protocol

To illustrate how our method allows us to optimize protocols that are genuinely adaptive, we consider here the following family of N-round games that we call *bit-sequence protocols*.

In the first round, we perform, with equal probability, each of the following measurements and record the outcome:

$$M_1 = \{M_{1|1} = |0\rangle\langle 0|, M_{2|1} = |1\rangle\langle 1|\},$$
$$M_2 = \{M_{1|2} = |+\rangle\langle +|, M_{2|2} = |-\rangle\langle -|\}. \quad (40)$$

In each of the $N - 1$ subsequent rounds, we then use the following procedure. In round k, we take the measurement outcome (1 or 2) of round $k - 1$ as the new measurement choice. We perform the respective measurement and record the new outcome. After the last measurement in round N, we check whether the final outcome is 2, in which case we assign a reward of 1; otherwise, the reward is 0.

For each N, the bit-sequence protocol defines an N-qubit operator that is applied to the N-qubit state of which we aim to certify its entanglement. In the following, we will consider an N-qubit state, ρ_N, that is an eigenstate to the maximal eigenvalue of this operator defined by the N-round protocol.
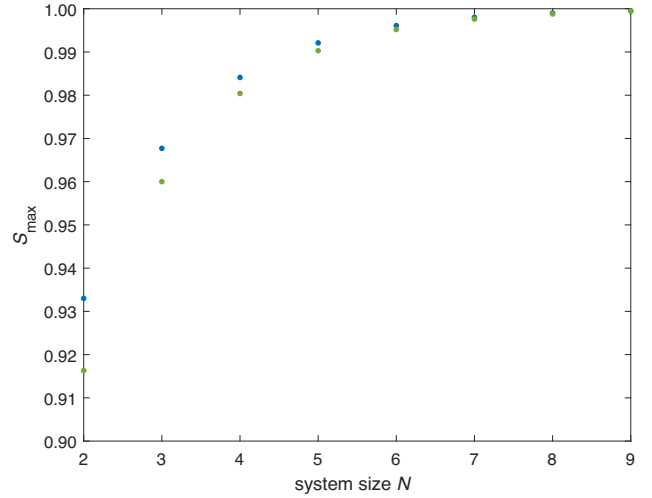

FIG. 10. Upper bounds for the winning probability of the best separable states compared to ρ_N in a single shot. The green dots give an upper bound on the score that can be obtained with separable states in the bit-sequence game. We observe that this value lies strictly below the value that we obtain with the states ρ_N (blue).

In Fig. 10, we compare the score, denoted $S_{\max}$, that we obtain with ρ_N with the maximum score that can be obtained with a separable state. The score for separable states is computed using the procedure from Sec. III (see Sec. IID of the Supplemental Material [49]).

In this protocol, the maximal score obtained with separable states corresponds—as in the protocol of Sec. IV B—to the maximal type-I error e_I, while the score for an entangled state ρ_N corresponds to $1 - e_{II}(\rho_N)$.

We observe a strict separation between the scores for ρ_N and the upper bounds that we obtain for separable states. This demonstrates that the bit-sequence protocols can certify the entanglement of this family of states. Our single-shot protocols can furthermore be turned into n-round protocols using, e.g., the ideas of metagames from Ref. [2], in order to amplify this separation.

V. ASYMPTOTIC BEHAVIOR OF SEQUENTIAL MODELS

In some scenarios, the considered sequential problem admits a natural generalization to arbitrarily many rounds or time steps N. In that case, we may also be interested in the asymptotic behavior of the total reward as $N \to \infty$. However, as we shall see in the following (Sec. V A), there are sequential models for which approximately computing the asymptotic maximum reward (or penalty) is an undecidable problem. Thus the best we can hope for are heuristics that work well in practice. In Sec. V B, we introduce such heuristic methods and we demonstrate in Sec. VI that these methods lead to useful bounds for the problems

that we have considered in the finite regime in Secs. IV A and IV B.

A. Sequential models with uncomputable asymptotics

To see that there are sequential models for which the asymptotic maximum reward cannot be well approximated, we rely on an intermediate result presented in Ref. [30], where specific probabilistic finite-state automata for which their asymptotic behavior cannot be well approximated have been constructed.

Consider a finite-state automaton $\mathcal{A}$ with a (finite) state and input sets Σ and X, initial state $\sigma_0 \in \Sigma$, and no outputs. The automaton is said to be *freezable* if there exists an input $x_0 \in X$ that keeps the automaton in the same internal state. Call some subset $A \subset \Sigma$ the set of *accepted states*. Then, we can consider how likely it is to bring the state of the automaton from σ_0 to some accepted state in n time steps by suitably choosing the input word $(x_1, \ldots, x_n) \in X^{\times n}$. Define

$$\text{val}_n(\mathcal{A}) = \max_{\bar{x} \in X^{\times n}} \text{Prob}\left(\sigma_n \in A\right). \tag{41}$$

The *value of the automaton* $\text{val}(\mathcal{A})$ is defined as $\sup_n \text{val}_n(\mathcal{A})$. Note that if $\mathcal{A}$ is freezable, then the function $\text{val}_n(\mathcal{A})$ is nondecreasing in n. In that case,

$$\text{val}(\mathcal{A}) = \lim_{n \to \infty} \text{val}_n(\mathcal{A}). \tag{42}$$

Given a freezable outputless automaton $\mathcal{A}$, consider the sequential model with S_k being the set of probability distributions over Σ, $H_k = X$, $\Lambda = \emptyset$, and $s_1 = \sigma_0$. The equation of motion is

$$P_{k+1}(\sigma') = \sum_{\sigma} P_k(\sigma) P_{\mathcal{A}}(\sigma'|\sigma, h), \tag{43}$$

where $P_{k+1} \in S_{k+1}, P_k \in S_k$, and $P_{\mathcal{A}}$ denotes the transition matrix of the automaton.

In a sequential problem with n rounds, we define the reward of the system as $r_k = 0 \ \forall \ k < n$, $r_n(s_n) = \sum_{\sigma \in A} P_n(\sigma)$. From all the above, it is clear that the maximum total reward of the system corresponds to $\nu_n^\star := \text{val}_n(\mathcal{A})$. Hence, $\lim_{n \to \infty} \nu_n^\star = \text{val}(\mathcal{A})$.

How difficult is to compute the asymptotic total reward for this kind of sequential models? The next result provides a very pessimistic answer [71].

Lemma 1 (Lemma 1 from Ref. [30]). For any rational $\lambda \in (0, 1]$, there exists a family $\mathcal{T}_\lambda$ of freezable probabilistic finite-state automata, with alphabet size $|X| = 5$ and with $|\Sigma| = 62$ states, such that, for all $\mathcal{A} \in T_\lambda$:

(a) $\text{val}(\mathcal{A}) \geq \lambda$ or $\text{val}(\mathcal{A}) \leq \lambda/2$
(b) it is undecidable which is the case

Now, consider the class $\mathcal{C}$ of sequential problems for which the asymptotic value exists and is contained in $[0, 1]$. This class contains the sequential problem of computing the value of any automaton in $\mathcal{T}_1$. It follows that there is no general algorithm to compute the asymptotic value of any sequential problem in $\mathcal{C}$ up to an error strictly smaller than $1/4$. Otherwise, one could use it to discriminate between the automata in $\mathcal{T}_1$ with value $1/2$ and those with value 1, in contradiction with the above lemma.

B. Heuristic methods for computing upper bounds on the asymptotics of sequential models

Upper bounds on the asymptotic value of a sequential model as $N \to \infty$ can be achieved by deriving a function $\nu(N)$ that upper bounds the solution of the corresponding sequential problem of size N for any N and taking the limit. A way to achieve this is to encode, via functional constraints, recursion relations that define value functions for arbitrarily high N. For simplicity of presentation, let us consider a sequential model where $r_k = 0$ for $k \neq N$ and f_k, S_k, and H_k are independent of k [72].

In the following, we show how to derive asymptotic bounds in case of convergence that is polynomial in $1/N$ as well as for exponential decay with N. Similar methods can also be used to bound the solution of the sequential problem even when the latter diverges, e.g., they can be used to prove that $\nu(N) \leq 4N^3$ (although, in this case, reducing the corresponding constraints to optimizations over functions of bounded variables might require some regularization trick (e.g., $V_k \to V_k/k^3$).

1. Sequential models with polynomial speed of convergence in 1/N

Let us assume that, for high N, $\nu(N) \approx \bar{\nu} + (A/N)$ and we wish to compute $\bar{\nu}$. This problem might arise, e.g., when we wish to study the asymptotic type-I error of a quantum preparation game [2]. To simplify the notation, we use negative indices $-j$ to denote $N - j$. Following the formulation in Eq. (12), in order to obtain an upper bound on the solution of the sequential problem for size N, we need to find $\nu, \{V_{-j}(s, \lambda)\}_{j=0}^{N-1}$ such that

$$V_0(s, \lambda) \geq r(s, h, \lambda), \tag{44}$$

$$V_{-(j+1)}(s, \lambda) \geq V_{-j}(f(s, h, \lambda), \lambda), \quad \forall j \in \{0, \ldots, N-1\} \tag{45}$$

$$\nu \geq V_{-(N-1)}(s_1, \lambda). \tag{46}$$

The value functions V_{-j} obtained according to Eq. (12) are generally different for each problem size N, so they do not help us solve the asymptotic problem.

Introducing an extra parameter into the value functions, in order to account for the increase in j, however, allows us to use the same value function from some k on. To see this,

suppose that we manage to show that there exist functions $\{W_{-j}(s, \lambda)\}_{j=0}^{k}$, $W(\alpha, s, \lambda)$ such that

$$W_0(s, \lambda) \geq r(s, \lambda, h), \tag{47}$$

$$W_{-j}(s, \lambda) \geq W_{-(j-1)}(f(s, h, \lambda), \lambda),$$
$$\forall j \in \{1, \ldots, k\}, \tag{48}$$

$$W\left(\frac{1}{k}, s, \lambda\right) \geq W_{-k}(s, \lambda), \tag{49}$$

$$W\left(\frac{\alpha}{\alpha+1}, s, \lambda\right) \geq W(\alpha, f(s, h, \lambda), \lambda), \quad \forall \alpha \in \left[0, \frac{1}{k}\right]. \tag{50}$$

Then, the functions

$$V_{-j}(s, \lambda) := \begin{cases} W_{-j}(s, \lambda), & j = 0, \ldots, k, \\ W\left(\frac{1}{j}, s, \lambda\right), & j = k+1, \ldots, N-1, \end{cases} \tag{51}$$

satisfy the constraints in Eqs. (44) and (45). Intuitively, in Eqs. (49) and (50), the variable α must be understood to represent $1/j$, in which case $\alpha/(\alpha+1)$ corresponds to $1/(j+1)$. The constraint in Eq. (50) hence implies the recursion relation Eq. (45) for all $j \geq k$, while the case $j < k$ is taken care of by Eq. (48). Note that the value of α is bounded; this is important if we wish to use polynomial optimization methods to enforce the inequality constraints in Eq. (50) (see Sec. VI).

Furthermore, if, for some real numbers $B, C, \ldots$, the extra condition

$$\bar{\nu} + A\alpha + B\alpha^2 + C\alpha^3 + \cdots \geq W(\alpha, s_1, \lambda),$$
$$\forall \alpha \in \left[0, \frac{1}{k}\right], \tag{52}$$

is satisfied, then, by Eq. (46), the function $\bar{\nu} + A\alpha + B\alpha^2 + C\alpha^3 + \cdots$ upper bounds the solution of the sequential problem. A heuristic to upper bound $\bar{\nu}^{\star}$ on the limit $\lim_{N \to \infty} \nu(N)$ thus consists in minimizing $\bar{\nu}$ under constraints [73] Eqs. (47)–(50) and (52).

The previous scheme is expected to lead to good upper bounds in situations in which, for high N, the optimal value functions $V_0, \ldots, V_{-(N-1)}$ of the problem given in Eq. (12) satisfy $V_{-(N-1)} \approx \bar{V} + (\tilde{V}/N)$, for some functions $\bar{V}, \tilde{V}$. That is, they work well as long as $\lim_{N \to \infty} V_{-(N-1)}$ exists. Experience with dynamical systems suggests, though, that even if the solution of the sequential problem tends to a given value $\bar{\nu}$, the sequence $(V_{-(N-1)})_N$ might asymptotically approach a *limit cycle*. This means that for many problems, only every $\mathcal{C}$th value function follows the required pattern: in such situations, there exist a natural number $\mathcal{C}$ (the period) and functions $V^0, \ldots, V^{\mathcal{C}-1}$ such that, for high enough N, $V_{-N} \approx V^R$, with $R = N \pmod{\mathcal{C}}$. Thus, instead of a single recursion relation [Eq. (50)], we need to construct a family of such relations.

This situation can thus be accounted for by considering functions $\{W_{-j}(s, \lambda)\}_{j=0}^{k}$, $\{W^j(\alpha, s, \lambda)\}_{j=0}^{\mathcal{C}-1}$ such that

$$W_0(s, \lambda) \geq r(s, \lambda, h),$$

$$W_{-j}(s, \lambda) \geq W_{-(j-1)}(f(s, h, \lambda), \lambda), \; j = 1, \ldots, k,$$

$$W^0\left(\frac{1}{k}, s, \lambda\right) \geq W_{-k}(s, \lambda),$$

$$W^1\left(\frac{\alpha}{\alpha+1}, s, \lambda\right) \geq W^0(\alpha, f(s, h, \lambda), \lambda), \quad \forall \alpha \in \left[0, \frac{1}{k}\right],$$

$$\vdots$$

$$W^{\mathcal{C}-1}\left(\frac{\alpha}{\alpha+1}, s, \lambda\right) \geq W^{\mathcal{C}-2}(\alpha, f(s, h, \lambda), \lambda), \; \forall \alpha \in \left[0, \frac{1}{k+\mathcal{C}-1}\right],$$

$$W^0\left(\frac{\alpha}{\alpha+1}, s, \lambda\right) \geq W^{\mathcal{C}-1}(\alpha, f(s, h, \lambda), \lambda), \; \forall \alpha \in \left[0, \frac{1}{k+\mathcal{C}}\right]. \tag{53}$$

2. Sequential models with exponential speed of convergence in N

Suppose that the speed of convergence of the sequential model is exponential, i.e., $\nu(N) \approx \bar{\nu} + A\gamma^N$, for $\gamma < 1$.

Assuming, for simplicity, that $\lim_{N \to \infty} V_{-(N-1)}$ exists, how could we estimate both $\bar{\nu}$ and γ?

One way to check that the postulated convergence rate holds is to verify the existence of functions $\{W_{-j}(s, \lambda)\}_{j=0}^{k}$,

$W(\alpha, s, \lambda)$ such that

$$W_0(s, \lambda) \geq r(s, \lambda, h),$$

$$W_{-j}(s, \lambda) \geq W_{-(j-1)}(f(s, h, \lambda), \lambda), \quad j = 1, \ldots, k,$$

$$W(\gamma^k, s, \lambda) \geq W_{-k}(s, \lambda),$$

$$W(\gamma\alpha, s, \lambda) \geq W(\alpha, f(s, \lambda, h), \lambda), \quad \forall \alpha \in [0, \gamma^k],$$
(54)

and Eq. (52) hold. The last line of the above equation is the induction constraint, wherein this time α is to be interpreted as γ^j. As before, we have defined α in terms of j such that the former quantity is bounded. To treat this problem within the formalism of convex optimization, one would try minimizing $\bar{\nu}$ for different values of γ. Each such minimization, if feasible, would result in an upper bound $\nu(N)$ on the solution of the sequential problem with $\nu(N) \approx \bar{\nu} + O(\gamma^N)$.

C. Adaptation to finite numerical precision

In standard numerical optimizations, the computer outputs value functions that only satisfy the positivity constraints in Eqs. (44)–(52) approximately. More generally, conditions of the form $\bullet \geq 0$ are weakened to $\delta + \bullet \geq 0$, for some $1 \gg \delta \geq 0$.

For small values of δ, the conditions in Eqs. (44)–(49) and (52) are not problematic, in the sense that a small infeasibility of any of these inequalities leads to a constant increase in the value function by that amount. This is why in the resolution of Eq. (12), we have so far not considered numerical precision issues explicitly. Indeed, let $\{\delta_j\}_{j=k+1}$ be such that

$$W_0(s, \lambda) + \delta_0 \geq r(s, \lambda, h),$$

$$W_{-j}(s, \lambda) + \delta_j \geq W_{-(j-1)}(f(s, h, \lambda), \lambda), \quad \forall j \in \{1, \ldots, k\}$$

$$W_{-k}(s, \lambda) \leq W\left(\frac{1}{k}, s, \lambda\right) + \delta_{k+1},$$

$$W(\alpha, s_1, \lambda) \leq \bar{\nu} + A\alpha + B\alpha^2 + C\alpha^2 + \cdots + \delta_\infty.$$
(55)

Then, the functions $\tilde{W}_{-j} := W_{-j} + \sum_{i=0}^{j} \delta_i$ and $\tilde{W} := W + \sum_{i=0}^{k+1} \delta_i$ satisfy conditions Eqs. (44)–(49). If, in addition, condition Eq. (50) is satisfied by the original functions, then

$$\bar{\nu} + \sum_{j=0}^{k+1} \delta_k + \delta_\infty$$
(56)

is a sound upper bound of the asymptotic value of the game.

However, precision issues become more problematic if the relation in Eq. (50) is only satisfied approximately, i.e., if

$$W\left(\frac{\alpha}{\alpha + 1}, s, \lambda\right) + \hat{\delta} \geq W(\alpha, f(s, h, \lambda), \lambda).$$
(57)

In that case, for $N > k$, the optimal reward of the Nth game round is upper bounded by

$$\bar{\nu} + \sum_{j=0}^{k+1} \delta_k + \delta_\infty + \hat{\delta}(N - k).$$
(58)

That is, the bound grows linearly with N. In that case, the asymptotic upper bounds $\bar{\nu}$ obtained numerically can only be considered valid for $N \ll O(1/\hat{\delta})$.

VI. APPLICATIONS OF ASYMPTOTIC OPTIMIZATION OF TIME-ORDERED PROCESSES

In this section, we bound the asymptotic behavior of the time-ordered processes that have been treated in a finite regime in Secs. IV A and IV B, respectively, relying on the techniques from Sec. V. While the former is an example of polynomial speed of convergence in $1/N$, the latter illustrates the technique in the exponential case.

A. Asymptotic behavior of finite-state automata

The probability that a two-state automaton generates a one-tick sequence of length L seems to converge to 0 as $O(1/L)$. It can also be verified that this is the convergence rate of the optimal model found in Ref. [23]. In fact, this model gives the probability of the one-tick sequence

$$p(2k) = \frac{1}{k}\left(1 - \frac{1}{k}\right)^{k-1} \approx \frac{1}{ek}, \quad \text{for } k \to \infty,$$
(59)

where $\mathbf{e}$ is the basis of natural logarithms and $p(2k - 1) = p(2k)$. This explicit example provides an upper bound on the converge rate for $\nu(L)$, i.e., $\nu(L)$ cannot converge to zero faster than $2/\mathbf{e}L$. Similarly to this model, the value $\nu(L)$ of the sequential game computed numerically satisfies $\nu(2k) = \nu(2k - 1)$ from $k = 2$ onward. This suggests that we should apply the relaxation Eq. (53), with $\mathcal{C} = 2$.

Remember that $s_k := (p_k(0), p_k(1))$, where $p_k(t)$ denotes the probability that the automaton has so far produced the outcome 0 and is in state t and $\lambda := (P(\sigma', 0|\sigma) : \sigma, \sigma')$, the transition matrix of the automaton. We allow the discrete-value functions $W_{-k}(s, \lambda)$ to be linear combinations of all monomials of the form $s^a \lambda^b$, with $a, b = 0, 1, 2$. The continuous-value functions W^1 and W^2 are linear combinations of the monomials $s^a \lambda^b \alpha^c$, with $a, b, c = 0, 1, 2$. Positivity constraints such as the last line of Eq. (53) are mapped to a polynomial problem by multiplying the whole expression by $(1 + \alpha)^2$.

To solve this problem, we have used the SDP solver MOSEK [52], in combination with YALMIP [56]. After

28 iterations, MOSEK returned the upper bound 0.0265, together with the unknown message. The solution found by MOSEK was, however, almost feasible, with parameters AS = 4×10^{-7}; DFEAS = 8.6×10^{-7} GFEAS = 7.5×10^{-11} and MU = 1.3×10^{-7}. It is therefore reasonable to assume [74] that the *actual* solution of the SDP, which for some reason MOSEK could not find, is indeed close to 2×10^{-2}. This figure is, indeed, greater than the conjectured null asymptotic value but nonetheless a nontrivial upper bound thereof.

B. Asymptotics of entanglement-detection protocols

We observe that for small N, $p_{\max}^{\text{sep}}$ behaves as

$$p_{\max}^{\text{sep}} \approx \frac{1}{2} + \frac{1}{2^N}. \tag{60}$$

This value of p^{sep} can also be achieved by preparing the separable state $|+\rangle^{\otimes N}$. For completeness, we illustrate this in Sec. III of the Supplemental Material [49]. This shows that our bounds from Fig. 8 for the finite-N regime are tight (up to numerical precision).

The behavior in Eq. (60) also implies that we expect an asymptotic value of $\frac{1}{2}$. Implementing the procedure to bound the asymptotic behavior from Sec. V B 2, we have been able to confirm an upper bound of 0.5004, before running into precision problems using YALMIP [56] and the solver MOSEK [52]. For the implementation, we have extended the one described in Sec. IIC of the Supplemental Material [49], with the additional relations from Sec. V B 2. Note that a value of $\frac{1}{2}$ corresponds to randomly guessing whether the state is entangled and is achievable for any N with the state Eq. (36). It is thus a lower bound on e_I. Note that gap between the success probability of 1 for the GHZ state and approximately $\frac{1}{2}$ for the worst-case separable state approximately reaches the maximal possible value of such a gap of $\frac{1}{2}$ [as explained in Eq. (37)]. This illustrates that the multiround GHZ game performs essentially optimally for the task of certifying the entanglement of a GHZ state.

VII. POLICIES WITH GUARANTEED MINIMUM REWARD

Let us complicate our sequential model a little bit, by adding a *policy*. The idea is that the parameters x describing our policy also affects the evolution of the system, i.e.,

$$s_{k+1} = f_k(s_k, h_k, \lambda, x) \tag{61}$$

as well as the reward $r_k(s_k, h_k, \lambda, x)$ obtained in each time step.

Ideally, we would like to find the policy x that minimizes the maximum penalty. That is, we wish to solve the problem

$$\min_x \max_{h,\lambda} \sum_{k=1}^{N} r_k(s_k, h_k, \lambda, x), \tag{62}$$

under the assumption that Eq. (61) holds and that we know the initial state s_1.

For fixed x, though, it might be intractable to compute the minimum expected reward. A more realistic and practical goal is thus to find a policy x and a sufficiently small upper bound ν on its maximum reward (or penalty) $\nu^\star(x)$. This is the subject of the next section.

A. Optimization over policies

For any x, let $\nu(x)$ denote an upper bound on $\nu^\star(x)$, obtained through some relaxation of the problem given in Eq. (12). We propose to use standard gradient-descent methods or some variant thereof, such as Adam [75], to identify a policy $x^\star$ with an acceptable guaranteed minimum reward $\nu(x^\star)$.

More concretely, call X the set of admissible policies, and assume that X is convex. Starting from some guess $x^{(0)}$ on the optimal policy, and given some decreasing sequence of non-negative values $(\epsilon_k)_k$, *projected gradient descent* works by applying the iterative equation

$$x^{(k+1)} = \pi_X \left(x^{(k)} - \epsilon_k \nabla_x \nu(x^{(k)}) \right), \tag{63}$$

where $\pi_Z(x)$ denotes the projection of vector x in set Z, i.e., the vector $z \in Z$ closest to x in Euclidean norm. For $k \gg 1$, we would expect policy $x^{(k)}$ to be close to minimal in terms of the guaranteed reward $\nu(x^{(k)})$.

The use of gradient methods requires estimating $\nabla_x \nu(\bar{x})$ for arbitrary policies $\bar{x}$. Considering that we have so far obtained bounds $\nu(x)$ (for fixed x) numerically from relaxations of Eq. (12), this task requires additional theoretical insight. The rest of this section is thus concerned with formulating an optimization problem to estimate the gradient of $\nu(x)$.

Suppose, for simplicity, that x consists of just one real parameter, i.e., we wish to compute $(d/dx)\nu(\bar{x})$. Fix the policy to be $\bar{x}$, let $\delta x > 0$, and let us consider the problem given in Eq. (12). To arrive at a tractable problem, one normally considers a simplification of the following type:

$$\nu(\bar{x}) := \min_{V_1,\ldots,V_N,\nu} \nu$$

$$\text{s.t. } V_N(s_N, \lambda) \geq_Q r_N(s_N, h, \lambda, \bar{x}),$$

$$V_k(s_k, \lambda) \geq_Q r_k(s_k, h, \lambda, \bar{x}) + V_{k+1}(f_{k+1}(s_k, h_k, \lambda), \lambda),$$

$$\nu \geq_Q V_1(s_1, \lambda). \tag{64}$$

Here, the relation $a \geq_Q b$ signifies that the expression $a - b$, evaluated on some convex superset Q of the set

of probability distributions on the variables $\{s_k, h_k\}_k, \lambda$, is non-negative. In the polynomial problems considered in this paper, this superset Q is, in fact, dual to the positivstellensatz used to enforce the positivity constraints. In the case of Putinar's positivstellensatz, Q would correspond to the set of monomial averages $\{\langle z_{i_1} \cdots z_{i_N} \rangle\}$ that, arranged in certain ways, define positive semidefinite matrices (for details, see Ref. [47]).

Since Q contains all probability distributions, $a \geq_Q b$ implies that $a \geq b$ and so the solution of Eq. (64) is a feasible point of Eq. (12) and hence an upper bound $\nu(\bar{x})$ on $\nu^\star(\bar{x})$. For the time being, we assume that the minimizer $V_1, \ldots, V_N$ of the problem given in Eq. (64) is unique for policy $\bar{x}$. We will drop this assumption later.

Now, let us consider how the problem given in Eq. (64) changes when we replace $\bar{x}$ by $\bar{x} + \delta x$. We assume not only that the solution satisfies $\nu(\bar{x} + \delta x) \approx \nu(\bar{x}) + \mu \delta x$ but also that the optimal slack variables $V_1, \ldots, V_N$ also experience an infinitesimal change, i.e.,

$$V_k(s_k, \lambda) \to V_k(s_k, \lambda) + \hat{V}_k(s_k, \lambda)\delta x. \quad (65)$$

That is, not only is the solution differentiable but also the minimizer of the optimization problem.

Ignoring all terms of the form $o(\delta x)$ in Eq. (65), it is clear that $\nu + \mu \delta x$ is an upper bound on $\nu^\star(\bar{x} + \delta x)$ if and only if

$$V_N(s_N, \lambda) + \hat{V}_N(s_N, \lambda)\delta x \geq_Q r_N(s_N, h, \lambda, \bar{x}) + \frac{\partial}{\partial x} r_N(s_N, h, \lambda, \bar{x})\delta x, \quad (66)$$

$$V_k(s_k, \lambda) + \hat{V}_k(s_k, \lambda)\delta x \geq_Q r_k(s_k, h, \lambda, \bar{x}) + \frac{\partial}{\partial x} r_k(s_k, h, \lambda, \bar{x})\delta x + V_{k+1}(s_{k+1}, \lambda)$$

$$+ \hat{V}_{k+1}(s_{k+1}, \lambda)\delta x + \delta x \sum_\sigma \frac{\partial V_{k+1}(s_{k+1}, \lambda)}{\partial s_{k+1}^\sigma} \frac{\partial f_k^\sigma(s_k, h_k, \lambda, x)}{\partial x}, \quad (67)$$

$$\nu + \mu \delta x \geq_Q V_1(s_1, \lambda) + \hat{V}_1(s_1, \lambda)\delta x. \quad (68)$$

Here, the symbol σ is used to denote the coordinates of s_{k+1}.

Our goal is now to turn this into an optimization problem that allows us to find μ. For this purpose, it is useful to invoke the fact that the slack variables $\{V_k\}_k$ are optimal, i.e., they not only satisfy the conditions in the problem given in Eq. (64) but all inequalities must be saturated. Namely, for each k, there exists $q \in Q$ such that

$$\langle V_k(s_k, \lambda) - r_k(s_k, h_k, \lambda) - V_{k+1}(s_{k+1}, \lambda)\rangle_q = 0. \quad (69)$$

To proceed, we need the following lemma, which allows us to phrase the constraints in Eqs. (66)–(68) in terms of the quantities $\hat{V}_k$.

Lemma 2. Let $a \geq_Q 0$, and let the set $Q' := \{q : q \in Q, \langle a \rangle_q = 0\}$ be nonempty. Then, $a + \hat{a}\delta x \geq_Q 0$ for $\delta x \to 0^+$ if and only if $\hat{a} \geq_{Q'} 0$.

Proof. $Q' \subset Q$, so $a + \hat{a}\delta x \geq_Q 0$ implies that $\hat{a} \geq_{Q'} 0$. To see the converse implication, note that, for $q \in Q \setminus Q'$, $\langle a \rangle_q > 0$ and therefore $\lim_{\delta x \to 0^+} \langle a + \hat{a}\delta x \rangle_q > 0$. To determine if $\langle a + \hat{a}\delta x \rangle_q \geq 0$ for all $q \in Q$ in the limit $\delta x \to 0^+$, it is thus sufficient to check that the property holds for $q \in Q'$, i.e., that $a \geq_{Q'} 0$. ∎

By duality, condition $\hat{a} \geq_{Q'} 0$ is equivalent to

$$\xi a + \hat{a} \geq_Q 0, \quad \text{for some } \xi \in \mathbb{R}. \quad (70)$$

From the above equation and the relations in Eq. (69), it follows that (under the hypothesis that the minimizer V is differentiable and unique at $\bar{x}$) the limit $D^+\nu(\bar{x}) \equiv \lim_{\delta x \to 0^+}(\nu(\bar{x} + \delta x) - \nu(\bar{x})/\delta x)$ equals

$$\min_{\mu, \hat{V}, \xi} \mu$$

such that $\hat{V}_N(s_k, \lambda) \geq_Q \frac{\partial}{\partial x} r_N(s_N, h_N, \lambda, \bar{x}) + \xi_N \left(V_N(s_k, \lambda) - r_N(s_N, h_N, \lambda, \bar{x}) \right),$

$$\hat{V}_k(s_k, \lambda) \geq_Q \frac{\partial}{\partial x} r_k(s_k, h, \lambda, \bar{x}) + \hat{V}_{k+1}(s_{k+1}, \lambda) + \sum_\sigma \frac{\partial V_{k+1}(s_{k+1}, \lambda)}{\partial s_{k+1}^\sigma} \frac{\partial f_k^\sigma(s_k, h_k, \lambda, x)}{\partial x} \quad (71)$$

$$+ \xi_k \left(V_k(s_k, \lambda) - r_k(s_k, h_k, \lambda, \bar{x}) - V_{k+1}(s_{k+1}, \lambda) \right),$$

$$\mu \geq_Q \hat{V}_1(s_1, \lambda) + \xi_1 \left(\nu - V_1(s_1, \lambda) \right).$$

If, in addition, the sets

$$Q_k' := \{q \in Q : q\,(V_k(s_k, \lambda) - r_k(s_k, h_k, \lambda, \bar{x}) - V_{k+1}(s_{k+1}, \lambda)) = 0\}, \quad k = 1, \dots, N,$$
$$Q_0' := \{q \in Q : q\,(\nu - V_1(s_1, \lambda)) = 0\} \tag{72}$$

have each cardinality 1, i.e., $Q_k' = \{q_k\}$, the problem to solve becomes even simpler, namely,

$$\min_{\mu, \hat{V}, \xi} \mu$$

$$\text{such that } q_N\left(\hat{V}_N(s_k, \lambda)\right) = q_N\left(\frac{\partial}{\partial x} r_N(s_N, h_N, \lambda, \bar{x})\right),$$

$$q_k\left(\hat{V}_k(s_k, \lambda)\right) = q_k\left(\frac{\partial}{\partial x} r_k(s_k, h, \lambda, \bar{x}) + \hat{V}_{k+1}(s_{k+1}, \lambda) + \sum_\sigma \frac{\partial V_{k+1}(s_{k+1}, \lambda)}{\partial s_{k+1}^\sigma} \frac{\partial f_k^\sigma(s_k, h_k, \lambda, x)}{\partial x}\right), \tag{73}$$

$$\mu = q_0\left(\hat{V}_1(s_1, \lambda)\right).$$

The derivation of Eq. (71) relies on the hypothesis that the optimal slack variables $V := \{V_k\}_k$ for policy $x = \bar{x}$ are unique. Should this not be true, the procedure to compute $D^+\nu(\bar{x})$ requires solving a nonconvex-optimization problem. The reader can find it in Sec. IV of the Supplemental Material [49], together with a heuristic to tackle it.

Now, let us return to the case of a multivariate policy x. If the gradient $\nabla_x \nu(\bar{x})$ exists, then its ith entry corresponds to the limit

$$\lim_{\delta x \to 0^+} \frac{\nu(\bar{x} + \delta x | i)) - \nu(\bar{x})}{\delta x}. \tag{74}$$

Each of these entries can be computed, in turn, via the procedure sketched above. This requires solving an optimization problem of complexity comparable to that of computing $\nu(\bar{x})$. Moreover, each such optimization can be performed separately for each coordinate of x. Namely, the process of computing the gradient of $\nu(\bar{x})$ can be *parallelized*. This allows us, through Eq. (63), to optimize over policies consisting of arbitrarily many parameters, as long as we have enough resources to compute $\nu(\bar{x})$.

VIII. APPLICATION: OPTIMIZATION OF ADAPTIVE PROTOCOLS FOR MAGIC STATE DETECTION

Magic states, i.e., states that cannot be expressed as convex combinations of stabilizer states, are a known resource for quantum computation: together with Clifford gates, they allow us to conduct universal quantum operations efficiently [76]. This raises the problem of certifying whether a given source of states is able to produce them [77].

In the qubit case, magic states are those that cannot be expressed as a convex combination of the eigenvectors of the three Pauli matrices. Equivalently, magic states are

those the Bloch vector $\vec{n}$ of which violates at least one of the inequalities

$$\left\{\sum_{j=1}^3 n_j a_j \leq 1 : a_1, a_2, a_3 \in \{-1, 1\}\right\} \tag{75}$$

(for an illustration, see Fig. 11).

We wish to devise an N-round adaptive measurement protocol that tells us whether a given source is preparing magic states. Namely, if the source can just produce nonmagic states, we expect the protocol to declare the source "magic" with low probability e_{I} (the type-I error of the protocol). If, however, the source is actually preparing independent copies of any quantum state violating one of these inequalities by an amount greater than or equal to
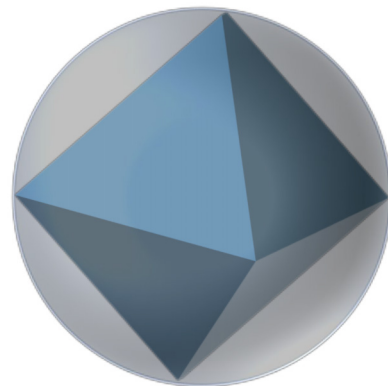

FIG. 11. Qubit magic states. Represented in the Bloch sphere, the magic states of a qubit are all those that lie outside the blue octahedron. In this section, the goal is to certify whether a general state is magic or not.
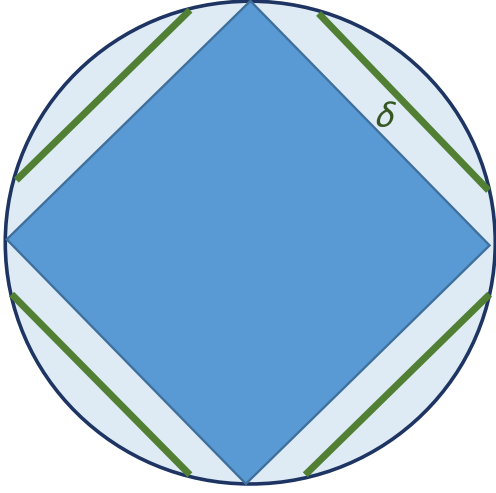
FIG. 12. An illustration of magic state detection. We display a cut through the Bloch sphere showing nonmagic (blue square) and magic states (region outside the square). The goal is to devise an N-round protocol that is able to reliably classify states into magic and nonmagic ones, as long as the considered magic states are at least δ away from the nonmagic ones. The protocol derived below, based on a two-state automaton, is valid in one of the regions outside the square of nonmagic states.

δ, we wish the protocol to declare the source "nonmagic" with low probability e_{II} (the type-II error of the protocol) (see Fig. 12).

To do so, we formulate the protocol as a time-independent preparation game [2]. That is, in every measurement round k, our knowledge of the magic of the state is encoded in the configuration $t \in T$ of the game, with $|T| < \infty$. The state prepared by the source is then measured by means of the POVM with elements $(M_{t'|t} : t' \in T) \subset B(\mathbb{C}^2)^{\times |T|}$. The result t' of this measurement is the new configuration of the game. When we exceed the total number of measurement rounds n, we must guess, based on the current game configuration, whether the state source is magic or not. For simplicity, we declare the source to be magic if $t \in A \subset T$, with $|A| = \lfloor |T|/2 \rfloor$. The initial state of the game, t_1, is chosen such that $t_1 \notin A$.

The game is thus defined by the POVMs $\{M_{t'|t} : t' \in T\}_t$. For computing e_I, we assume that the "dishonest" player, who claims to produce magic states but does not, knows at every round the current game configuration. They can thus adapt their state preparation correspondingly to increase the type-I error. As explained in Sec. III B, the computation of the maximum score of a preparation game under adaptive strategies can be cast as a sequential problem with $h_k := (\rho_k(t) : t \in T_k)$. In this case, the set of feasible states, C, is generated by convex combinations of $\{\psi_i\}_{i=1}^6$, the eigenvectors of the three Pauli matrices: linear maximizations over C therefore correspond to maximizations over these six "vertices." Putting it all together, we find

that computing the type-I error of the game amounts to applying the recursion relation:

$$\mu_N(t) = \max_i \sum_{t' \in A} \mathrm{tr}(|\psi_i\rangle\langle\psi_i|M_{t'|t}),$$

$$\mu_k(t) = \max_i \sum_{t'} \mathrm{tr}(|\psi_i\rangle\langle\psi_i|M_{t'|t})\mu_{k+1}(t'), \qquad (76)$$

$$e_I = \mu_1(t_1).$$

Let us assume that all the maximizations above have a unique maximizer and call $i(k, t)$ the maximizer corresponding to the kth round and game configuration t. Then, the gradient of e_I with respect to the POVM element $M_{\tau'|\tau}$ satisfies the recursion relations:

$$\vec{\nabla}\mu_N(t) = \delta_{\tau,t}|\psi_{i(N,\tau)}\rangle\langle\psi_{i(N,\tau)}|,$$

$$\vec{\nabla}\mu_k(t) = \sum_{t'} \mathrm{tr}(|\psi_{i(k,t)}\rangle\langle\psi_{i(k,t)}|M_{t'|t})\vec{\nabla}\mu_{k+1}(t')$$

$$+ \delta_{\tau,t}\mu_{k+1}(\tau')|\psi_{i(k,\tau)}\rangle\langle\psi_{i(k,\tau)}|,$$

$$\vec{\nabla}e_I = \vec{\nabla}\mu_1(t_1). \qquad (77)$$

It can hence be computed efficiently in the number of measurement rounds.

Now, consider the computation of the type-II error under IID strategies, the formulation of which as a sequential problem can be found in Sec. II C. We first focus on the set $\mathcal{S}_a$ of qubit states with Bloch vector $\vec{n}$ satisfying $\sum_j a_j n_j \geq 1 + \delta$, for some $a_1, a_2, a_3 \in \{-1, 1\}$. Applied to a source that always produces the same state $\rho \in \mathcal{S}$, the magic detection protocol can be modeled through a sequential model, with internal state s_k at time k given by the probability distribution over the game configurations $P_k(t)$ in round k, just before the measurement. The state $\rho \in \mathcal{S}$ prepared by the source is to be identified with the evolution parameters λ, as the equation of motion of the model is

$$P_{k+1}(t') = \sum_t \mathrm{tr}(\rho M_{t'|t})P_k(t). \qquad (78)$$

To calculate the type-II error, we assign the model the rewards $r_k = 0$, for $k = 1, \ldots, N-1$ and

$$r_N(P_N, \rho) = \sum_t P_N(t) \sum_{t' \in A} \mathrm{tr}(\rho M_{t'|t}). \qquad (79)$$

We can thus compute an upper bound on $e_{II}(\mathcal{S}_a)$, the maximum type-II error achieved with states in class $\mathcal{S}_a$, via the SDP relaxation of Eq. (12), and its gradient with respect to the POVM element $M_{t'|t}$, via Eq. (71) [78]. We assume that the solution of problem given in Eq. (12) is unique, so that the techniques of Sec. IV of the Supplemental Material [49] are not necessary.
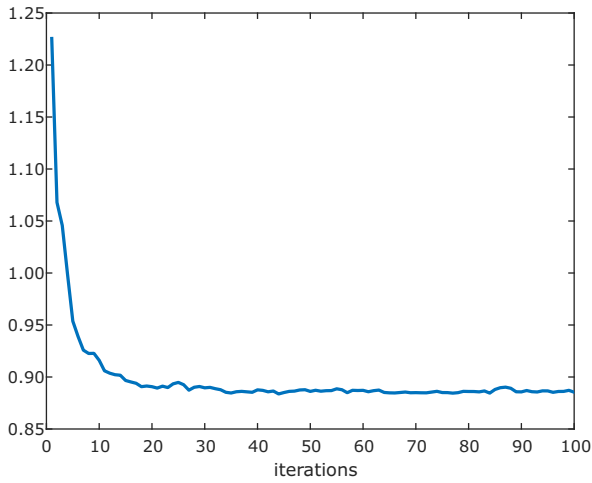
FIG. 13. The upper bounds for $e_I + e_{II}$ as a function of the number of gradient iterations. Starting from a random preparation game, we have used the Adam algorithm [75] to decrease the sum of the type-I and type-II errors of a magic state detection protocol. For this computation, we set the learning rate of the algorithm to 0.01.

Our original problem, though, is to compute the type-II error of all states violating at least one of the facets of the magic polytope by an amount $\geq \delta$. Thus we have that $e_{II} = \max_a e_{II}(\mathcal{S}_a)$. The gradient (or, more properly said, subgradient) of this function is $\vec{\nabla} e_{II}(\mathcal{S}_{a^\star})$, where $a^\star$ is the argument of $\max_a e_{II}(\mathcal{S}_a)$.

We are now ready to apply gradient descent to minimize the combined error $e_I + e_{II}$. We have used the gradient method Adam [75], followed at each iteration by a projection onto the (convex) set of protocols $\{M_{t'|t} : t' \in T\}_{t \in T}$, with $M_{t'|t} \geq 0$, $\forall t, t' \in T, \sum_{t'} M_{t'|t} = \mathbb{I}$, $\forall t \in T$. For $|T| = 2$, $n = 6$, gradient descent converges to a value $e_I + e_{II} \approx 1$, probably an indication that one cannot detect all magic states with a two-state automaton. If, however, we restrict the problem to that of detecting those states that satisfy the inequality $\sum_i n_i \geq 1 + \delta$, we arrive at the plot shown in Fig. 13.

As the reader can appreciate, after a few iterations, the algorithm successfully identifies a strategy with $e_I + e_{II} < 1$. Beyond 50 iterations, the objective value does not seem to decrease much further. This curve has to be understood as a proof of principle that gradient methods are useful to devise preparation games. Note that for a general preparation game, the number of components of the gradient is $4|T|^2$. Optimizations over preparation games with high $|T|$ would thus benefit from the use of parallel processing to compute the whole gradient vector.

IX. CONCLUSIONS

We have proposed a method to relax sequential problems involving a finite number of rounds. This method allows us to solve optimization problems of sizes that have not been tractable with previous convex-optimization techniques. Moreover, a variant of the method generates upper bounds on the solution of the problem in the asymptotic limit of infinitely many rounds.

We have demonstrated the practical use of our methods by solving an open problem: namely, we have computed the maximum probability for any finite-state automaton of a certain fixed dimension to produce certain sequences, thus proving that the values conjectured in Refs. [23,24] are optimal (up to numerical precision). We have further illustrated how to turn such bounds into analytical results.

As we have shown, the methods are also relevant in the context of certifying properties of quantum systems, especially in entanglement detection. Specifically, our method enables us to bound the misclassification error for separable states in multiround schemes as well as in the asymptotic limit, which we illustrate by means of a GHZ-game-inspired protocol, for which we also show that our bounds are tight.

Finally, we have explained how to combine these methods with gradient-descent techniques to optimize over sequential models with a certified maximum reward. This has allowed the computer to generate an adaptive protocol for magic state detection.

In some of our optimization problems, we have found that otherwise successful SDP solvers, such as MOSEK [52], have failed in some cases to output a reliable result. The cause of this atypical behavior is unclear and we lack a general solution for this issue. That said, the results from the numerical optimization have still allowed us to extract and confirm reliable bounds from the optimizations in most cases.

In two of the considered applications, we have found tight upper bounds on the quantities of interest. This may indicate that, at least in reasonably simple cases, our hierarchical method converges at relatively low levels of complexity. Finding conditions for convergence at such low levels is an interesting question that we leave open.

Quantum technologies are currently at the verge of building computing devices that can be used beyond purely academic purposes. These involve more than a few particles and thus the problem of certifying properties of systems of intermediate sizes is currently crucial. This is relevant not only from the perspective of a company building such devices but also from the perspective of a user who may want to test them independently. This progress is somewhat in tension with the state of research in certification, e.g., in entanglement theory, where methods are only abundant in the few-particle regime. Our GHZ game illustrates that automaton-guided protocols may bridge this gap: modeling one-way LOCC protocols through sequential processes might allow the certification of large classes of many-body systems, thus offering an alternative to protocols based on shadow tomography [65,79].

Our methods are furthermore able to certify systems beyond the linear regime that is employed when considering the usual types of witnesses for certifying systems, which from a mathematical viewpoint essentially distinguish convex sets by finding a separating hyperplane. As suggested in the example of magic states, our techniques beyond this regime also mean that we can analyze more general classes of systems simultaneously. Indeed, using a finite-state automaton of large enough dimension, we expect to be able to detect any states that have at least a certain distance from the set of nonmagic states. As optimization methods improve, we expect such *universal* protocols to become more and more common, as they are also applicable with minimal knowledge about the state of the system at hand.

Finally, many different problems encountered in quantum information theory have a sequential structure similar to the one analyzed in this paper. Thus we expect that our methods will soon find applications beyond those studied in the present work. In this regard, it would be interesting to adapt our techniques to analyze quantum communication protocols. Another natural research line would be to extend our methods to model interactions with an evolving quantum system, rather than a (classical) finite state automaton.

ACKNOWLEDGMENTS

---

[1] A. Wallraff, D. I. Schuster, A. Blais, L. Frunzio, J. Majer, M. H. Devoret, S. M. Girvin, and R. J. Schoelkopf, Approaching unit visibility for control of a superconducting qubit with dispersive readout, Phys. Rev. Lett. **95**, 060501 (2005).

[2] M. Weilenmann, E. A. Aguilar, and M. Navascués, Analysis and optimization of quantum adaptive measurement protocols with the framework of preparation games, Nat. Commun. **12**, 4553 (2021).

[3] S. Geller, D. C. Cole, S. Glancy, and E. Knill, Improving quantum state detection with adaptive sequential observations, ArXiv:2204.00710.

[4] M. Navascués, S. Pironio, and A. Acín, Bounding the set of quantum correlations, Phys. Rev. Lett. **98**, 010401 (2007).

[5] M. Navascués, S. Pironio, and A. Acín, A convergent hierarchy of semidefinite programs characterizing the set of quantum correlations, New J. Phys. **10**, 073013 (2008).

[6] T. Moroder, J.-D. Bancal, Y.-C. Liang, M. Hofmann, and O. Gühne, Device-independent entanglement quantification and related applications, Phys. Rev. Lett. **111**, 030501 (2013).

[7] V. Saggio, A. Dimić, C. Greganti, L. A. Rozema, P. Walther, and B. Dakić, Experimental few-copy multipartite entanglement detection, Nat. Phys. **15**, 935 (2019).

[8] A. Paz, *Introduction to Probabilistic Automata* (Academic Press, 1971). https://www.sciencedirect.com/book/978012 5476508/introduction-to-probabilistic-automata.

[9] M. O. Rabin, Probabilistic automata, Inf. Control **6**, 230 (1963).

[10] M. Kleinmann, O. Gühne, J. R. Portillo, J. Åke Larsson, and A. Cabello, Memory cost of quantum contextuality, New J. Phys. **13**, 113011 (2011).

[11] G. Fagundes and M. Kleinmann, Memory cost for simulating all quantum correlations from the Peres-Mermin scenario, J. Phys. A **50**, 325302 (2017).

[12] C. Budroni, G. Fagundes, and M. Kleinmann, Memory cost of temporal correlations, New J. Phys. **21**, 093018 (2019).

[13] C. Budroni, A. Cabello, O. Gühne, M. Kleinmann, and J.-A. Larsson, Kochen-Specker contextuality, Rev. Mod. Phys. **94**, 045007 (2022).

[14] A. J. P. Garner, Q. Liu, J. Thompson, V. Vedral, and M. Gu, Provably unbounded memory advantage in stochastic simulation using quantum mechanics, New J. Phys. **19**, 103009 (2017).

[15] T. J. Elliott and M. Gu, Superior memory efficiency of quantum devices for the simulation of continuous-time stochastic processes, npj Quantum Inf. **4**, 1 (2018).

[16] T. J. Elliott, C. Yang, F. C. Binder, A. J. P. Garner, J. Thompson, and M. Gu, Extreme dimensionality reduction with quantum modeling, Phys. Rev. Lett. **125**, 260501 (2020).

[17] C. Spee, Certifying the purity of quantum states with temporal correlations, Phys. Rev. A **102**, 012420 (2020).

[18] J. Hoffmann, C. Spee, O. Gühne, and C. Budroni, Structure of temporal correlations of a qubit, New J. Phys. **20**, 102001 (2018).

[19] C. Spee, H. Siebeneich, T. F. Gloger, P. Kaufmann, M. Johanning, M. Kleinmann, C. Wunderlich, and O. Gühne, Genuine temporal correlations can certify the quantum dimension, New J. Phys. **22**, 023028 (2020).

[20] C. Spee, C. Budroni, and O. Gühne, Simulating extremal temporal correlations, New J. Phys. **22**, 103037 (2020).

[21] P. Erker, M. T. Mitchison, R. Silva, M. P. Woods, N. Brunner, and M. Huber, Autonomous quantum clocks: Does thermodynamics limit our ability to measure time? Phys. Rev. X **7**, 031022 (2017).

[22] M. P. Woods, R. Silva, G. Pütz, S. Stupar, and R. Renner, Quantum clocks are more accurate than classical ones, PRX Quantum **3**, 010319 (2022).

[23] C. Budroni, G. Vitagliano, and M. P. Woods, Ticking-clock performance enhanced by nonclassical temporal correlations, Phys. Rev. Res. **3**, 033051 (2021).

[24] L. B. Vieira and C. Budroni, Temporal correlations in the simplest measurement sequences, Quantum **6**, 623 (2022).

[25] M. Zurel, C. Okay, and R. Raussendorf, Hidden variable model for universal quantum computation with magic states on qubits, Phys. Rev. Lett. **125**, 260404 (2020).

[26] G. Vitagliano and C. Budroni, Leggett-Garg macrorealism and temporal correlations, Phys. Rev. A **107**, 040101 (2023).

[27] F. G. Brandao, A. W. Harrow, J. R. Lee, and Y. Peres, in *Proceedings of the 5th Conference on Innovations in Theoretical Computer Science* (2014), p. 183.

[28] Y. Li, V. Y. Tan, and M. Tomamichel, Optimal adaptive strategies for sequential quantum hypothesis testing, Commun. Math. Phys. **392**, 993 (2022).

[29] H. Gimbert and Y. Oualhadj, in *International Colloquium on Automata, Languages, and Programming* (Springer, 2010), p. 527. https://link.springer.com/chapter/10.1007/978-3-642-14162-1_44.

[30] D. Elkouss and D. Pérez-García, Memory effects can make the transmission capability of a communication channel uncomputable, Nat. Commun. **9**, 1 (2018).

[31] M. P. Woods, R. Silva, and J. Oppenheim, Autonomous quantum machines and finite-sized clocks, Ann. Henri Poincaré **20**, 125 (2019).

[32] Y. Yang, L. Baumgärtner, R. Silva, and R. Renner, Accuracy enhancing protocols for quantum clocks, ArXiv:1905.09707.

[33] Y. Yang and R. Renner, Ultimate limit on time signal generation, ArXiv:2004.07857.

[34] E. Schwarzhans, M. P. E. Lock, P. Erker, N. Friis, and M. Huber, Autonomous temporal probability concentration: Clockworks and the second law of thermodynamics, Phys. Rev. X **11**, 011046 (2021).

[35] F. Meier, E. Schwarzhans, P. Erker, and M. Huber, Fundamental accuracy-resolution trade-off for timekeeping devices, Phys. Rev. Lett. **131**, 220201 (2023).

[36] R. Silva, N. Nurgalieva, and H. Wilming, Ticking clocks in quantum theory, ArXiv:2306.01829.

[37] A. N. Pearson, Y. Guryanova, P. Erker, E. A. Laird, G. A. D. Briggs, M. Huber, and N. Ares, Measuring the thermodynamic cost of timekeeping, Phys. Rev. X **11**, 021029 (2021).

[38] It does not matter which of the initial states is chosen, as these choices are equivalent up to relabeling.

[39] S. Gharibian, Strong NP-hardness of the quantum separability problem, Quantum Inf. Comput. **10**, 343 (2010).

[40] I. P. McCulloch, Infinite size density matrix renormalization group, revisited, ArXiv:0804.2509.

[41] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming* (John Wiley & Sons, 2014). https://www.wiley.com/en-us/Markov+Decision+Processes%3A+Discrete+Stochastic+Dynamic+Programming-p-9781118625873.

[42] J. B. Lasserre, D. Henrion, C. Prieur, and E. Trélat, Nonlinear optimal control via occupation measures and LMI-relaxations, SIAM J. Control Optim. **47**, 1643 (2008).

[43] J. L. Krivine, Anneaux préordonnés, J. Anal. Math. **12**, 307 (1964).

[44] J. L. Krivine, A nullstellensatz and a positivstellensatz in semialgebraic geometry, Math. Ann. **207**, 87 (1974).

[45] K. Schmüdgen, The K-moment problem for compact semialgebraic sets, Math. Ann. **289**, 203 (1991).

[46] M. Putinar, Positive polynomials on compact semialgebraic sets, Indiana Univ. Math. J. **42**, 969 (1993).

[47] J. B. Lasserre, Global optimization with polynomials and the problem of moments, SIAM J. Optim. **11**, 796 (2001).

[48] P. Parrilo, Semidefinite programming relaxations for semialgebraic problems, Math. Program. Ser. B **96**, 293 (2003).

[49] See the Supplemental Material at http://link.aps.org/supplemental/10.1102 which contains a brief description of the Lasserre-Parrilo hierarchy and variants, and the details of the implementation of optimization procedures.

[50] R. Murray, V. Chandrasekaran, and A. Wierman, Newton polytopes and relative entropy optimization, Found. Comput. Math. **21**, 1703 (2021).

[51] M. Dressler and R. Murray, Algebraic perspectives on signomial optimization, ArXiv:2107.00345.

[52] M. ApS, *The* MOSEK *Optimization Toolbox for* MATLAB *Manual. Version 9.0.* (2019).

[53] S. Diamond and S. Boyd, CVXPY: A PYTHON-embedded modeling language for convex optimization, J. Mach. Learn. Res. **17**, 1 (2016).

[54] B. O'Donoghue, E. Chu, N. Parikh, and S. Boyd, SCS: Splitting conic solver, version 3.2.2, https://github.com/cvxgrp/scs (2022).

[55] Recall that the variables of the polynomial optimization are the coefficients of the V_k rather than the variables of the sequential model s_k.

[56] J. Löfberg, in *Proceedings of the CACSD Conference* (Taipei, Taiwan, 2004).

[57] Note that SOLVEMOMENT sometimes allows optimal solutions to the problem to be directly extracted, as is the case in the examples below. This is not surprising, as in cases in which the Lasserre hierarchy has converged at a given finite level, the optimal values of the polynomial variables can be found in the moment matrix.

[58] Deriving this upper bound analytically, is straightforward if we require in addition that $P(b = 1|0) = 0$, which we can impose without changing our upper bounds. In this case, the polynomials simplify in a way that inspecting the gradient immediately leads to these bounds.

[59] Note that we have changed the notation for value functions from $\{V_k\}_k$ to $\{W_k\}_k$ here, to make the distinction from the $L = 3$ case clear.

[60] That $W_2(s_2, \lambda)$ upper bounds the reward follows as in the $L = 3$ case above.

[61] Analytically, checking this is again more straightforward when assuming additionally that $P(b = 1|0) = 0$.

[62] K. X. Wei, I. Lauer, S. Srinivasan, N. Sundaresan, D. T. McClure, D. Toyli, D. C. McKay, J. M. Gambetta, and S. Sheldon, Verifying multipartite entangled Greenberger-Horne-Zeilinger states via multiple quantum coherences, Phys. Rev. A **101**, 032343 (2020).

[63] C. Song, K. Xu, H. Li, Y.-R. Zhang, X. Zhang, W. Liu, Q. Guo, Z. Wang, W. Ren, J. Hao, H. Feng, H. Fan, D. Zheng, D.-W. Wang, H. Wang, and S.-Y. Zhu, Generation of multicomponent atomic Schrödinger cat states of up to 20 qubits, Science **365**, 574 (2019).

[64] A. Omran, H. Levine, A. Keesling, G. Semeghini, T. T. Wang, S. Ebadi, H. Bernien, A. S. Zibrov, H. Pichler, S. Choi, J. Cui, M. Rossignolo, P. Rembold, S. Montangero, T. Calarco, M. Endres, M. Greiner, V. Vuletić, and M. D. Lukin, Generation and manipulation of Schrödinger cat states in Rydberg atom arrays, Science **365**, 570 (2019).

[65] H.-Y. Huang, R. Kueng, and J. Preskill, Predicting many properties of a quantum system from very few measurements, Nat. Phys. **16**, 1050 (2020).

[66] O. Gühne and G. Tóth, Entanglement detection, Phys. Rep. **474**, 1 (2009).

[67] N. D. Mermin, Extreme quantum entanglement in a superposition of macroscopically distinct states, Phys. Rev. Lett. **65**, 1838 (1990).

[68] M. Boyer, Extended GHZ n-player games with classical probability of winning tending to 0, ArXiv:quant-ph/0408090.

[69] Given the small numbers of experimental rounds, we used the mean instead of the "median of means" recommended in Ref. [65] to estimate the GHZ fidelity in the shadow protocols.

[70] S. T. Flammia and Y.-K. Liu, Direct fidelity estimation from few Pauli measurements, Phys. Rev. Lett. **106**, 230501 (2011).

[71] The statement in Ref. [30] is more general in the sense that the lemma is proven when additionally requiring the automata to be resettable. This property is, however, not required for our application and we thus omit it from the statement below.

[72] Note that this restriction is not necessary. In particular, analogous methods can be used to deal with scenarios in which the f_k and r_k are k dependent, as long as we can formulate analogous constraints in terms of polynomials of the corresponding bounded variable α below.

[73] It is advisable, on the grounds of numerical stability, to also restrict the ranges of possible values for $A, B, C, \ldots$.

[74] To further support the conjecture that the value reported by MOSEK is close to the actual solution of the SDP, we have considered the value functions $\{W_{-j}\}_{j=0}^3$, W^1, and W^2 output by the solver and then used the Lasserre hierarchy to verify that the relations in Eq. (53) hold. We have only found tiny violations of positivity (of order -10^{-5}) for the last two recursion relations.

[75] D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, ArXiv:1412.6980.

[76] S. Bravyi and A. Kitaev, Universal quantum computation with ideal Clifford gates and noisy ancillas, Phys. Rev. A **71**, 022316 (2005).

[77] V. Veitch, S. A. H. Mousavian, D. Gottesman, and J. Emerson, The resource theory of stabilizer quantum computation, New J. Phys. **16**, 013009 (2014).

[78] The dual problem of Eq. (12) does not seem to have a unique solution, because we find that the use of Eq. (73) results in infeasible semidefinite programs.

[79] S. Aaronson, in *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing* (2018), p. 325.