



Rate-independent Computation in Continuous Chemical Reaction Networks

HO-LIN CHEN, National Taiwan University, Taiwan

DAVID DOTY, University of California, Davis, USA

WYATT REEVES, Harvard University, USA

DAVID SOLOVEICHIK, University of Texas at Austin, USA

Understanding the algorithmic behaviors that are *in principle* realizable in a chemical system is necessary for a rigorous understanding of the design principles of biological regulatory networks. Further, advances in synthetic biology herald the time when we will be able to rationally engineer complex chemical systems and when idealized formal models will become blueprints for engineering.

Coupled chemical interactions in a well-mixed solution are commonly formalized as chemical reaction networks (CRNs). However, despite the widespread use of CRNs in the natural sciences, the range of computational behaviors exhibited by CRNs is not well understood. Here, we study the following problem: What functions $f : \mathbb{R}^k \rightarrow \mathbb{R}$ can be computed by a CRN, in which the CRN eventually produces the correct amount of the “output” molecule, no matter the rate at which reactions proceed? This captures a previously unexplored but very natural class of computations: For example, the reaction $X_1 + X_2 \rightarrow Y$ can be thought to compute the function $y = \min(x_1, x_2)$. Such a CRN is robust in the sense that it is correct whether its evolution is governed by the standard model of mass-action kinetics, alternatives such as Hill-function or Michaelis-Menten kinetics, or other arbitrary models of chemistry that respect the (fundamentally digital) stoichiometric constraints (what are the reactants and products?).

We develop a reachability relation based on a broad notion of “what could happen” if reaction rates can vary arbitrarily over time. Using reachability, we define *stable computation* analogously to probability 1 computation in distributed computing and connect it with a seemingly stronger notion of rate-independent computation based on convergence in the limit $t \rightarrow \infty$ under a wide class of generalized rate laws. Besides the direct mapping of a concentration to a nonnegative analog value, we also consider the “dual-rail representation” that can represent negative values as the difference of two concentrations and allows the composition of CRN modules. We prove that a function is rate-independently computable if and only if it is piecewise linear (with rational coefficients) and continuous (dual-rail representation), or non-negative with discontinuities occurring only when some inputs switch from zero to positive (direct representation). The many contexts where continuous piecewise linear functions are powerful targets for implementation, combined with the systematic construction we develop for computing these functions, demonstrate the potential of rate-independent chemical computation.

D. Doty was supported by NSF grants 2211793, 1900931, and 1844976. D. Soloveichik was supported by NSF grants CCF-1901025, CCF-1652824, and a Sloan Foundation Research Fellowship. W. Reeves was supported by the National Science Foundation Graduate Research Fellowship under Grant No. DGE1745303. H.-L. Chen was supported by MOST (Taiwan) grants 107-2221-E-002-031-MY3 and 110-2223-E-002-006-MY3.

Authors’ addresses: H.-L. Chen, National Taiwan University, MD718, No. 1, Sec. 4, Roosevelt Rd., Taipei 106216, Taiwan; email: holinchen@ntu.edu.tw; D. Doty, University of California, Davis, One Shields Ave, Davis, CA 95616, USA; email: doty@ucdavis.edu; W. Reeves, Harvard University, 1 Oxford St, Cambridge, MA 02138, USA; email: wreeves@math.harvard.edu; D. Soloveichik, University of Texas at Austin, 2501 Speedway, Austin, TX 78712, USA; email: david.soloveichik@utexas.edu.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

© 2023 Copyright held by the owner/author(s).

0004-5411/2023/05-ART22 \$15.00

<https://doi.org/10.1145/3590776>

CCS Concepts: • **Computer systems organization** → **Analog computers**; • **Theory of computation** → **Models of computation**;

Additional Key Words and Phrases: Chemical reaction networks, mass-action, analog computation, piecewise-linear

ACM Reference format:

Ho-Lin Chen, David Doty, Wyatt Reeves, and David Soloveichik. 2023. Rate-independent Computation in Continuous Chemical Reaction Networks. *J. ACM* 70, 3, Article 22 (May 2023), 61 pages.
<https://doi.org/10.1145/3590776>

1 INTRODUCTION

Understanding the dynamic behaviors that are, in principle, achievable with chemical species interacting over time is crucial for engineering of complex molecular systems capable of diverse and robust behaviors. The exploration of this space also helps to elucidate the constraints imposed upon biology by the laws of chemistry. The natural language for describing the interactions of molecular species in a well-mixed solution is that of **chemical reaction networks (CRNs)**, i.e., finite sets of chemical reactions such as $A + B \rightarrow A + C$. The intuitive meaning of this expression is that a unit of chemical species A reacts with a unit of chemical species B , producing a unit of a new chemical species C and regenerating a unit of A back. Typically (in mass-action kinetics) the rate with which this occurs is proportional to the product of the amounts of the reactants A and B .

Informally speaking, we can identify two sources of computational power in CRNs. First, the reaction *stoichiometry* transforms some specific ratios of reactants to products. For example, $X \rightarrow 2Y$ makes two units of Y for every unit of X . Second, in mass-action kinetics the reaction *rate laws* effectively perform multiplication of the reactant concentrations. In this work, we seek to disentangle the contributions of these two computational ingredients by focusing on the computational power of stoichiometry alone. Besides fundamental scientific interest, such rate-independent computation may be significantly easier to engineer than computation relying on rates (see Section 1.2). Importantly, stoichiometry is robust—not requiring the tuning of reaction conditions, nor even the assumption that the solution is well-mixed.

In the *discrete* model of chemical kinetics (see Section 1.2 for the distinction between the discrete and continuous models), rate-independence is formally related to probability 1 computation with passively mobile (i.e., interacting randomly) agents in distributed computing (the “population protocols” model [6, 9]; see Section 1.3). However, the continuous model of chemistry is most widely used and is more applicable for engineering chemical computation where working with bulk concentrations remains the state-of-the-art (see Section 1.2). This article formally articulates rate-independence in continuous CRNs and characterizes the computational power of stoichiometry in this model.

In the continuous setting, the amount of a species is a nonnegative real number representing its concentration (average count per unit volume).¹ We characterize the class of real-valued functions computable by CRNs when reaction rates are permitted to vary arbitrarily (possibly adversarially) over time. Any computation in this setting must rely on stoichiometry alone. How can rate laws “preserve stoichiometry” while varying “arbitrarily over time”? Formally, preserving stoichiometry means that if we reach state \mathbf{d} from state \mathbf{c} , then $\mathbf{d} = \mathbf{c} + \mathbf{M}\mathbf{u}$ for some non-negative vector \mathbf{u} of reaction fluxes, where the CRN’s stoichiometry matrix \mathbf{M} maps those fluxes to the changes in species concentrations they cause. (For example, flux 0.5 of reaction $C + X \rightarrow C + 3Y$ changes the

¹Although the finite density of matter physically restricts what the largest concentration of any species could realistically be, standard models of chemical kinetics focus on systems that are far from this bound, mathematically allowing concentrations to be arbitrarily large.

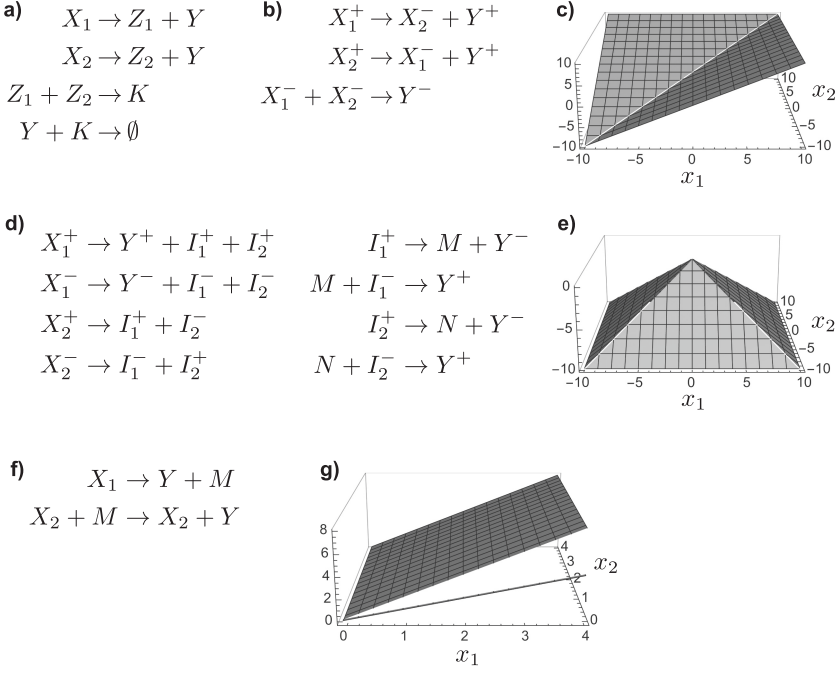


Fig. 1. Examples of rate-independent computation with chemical reaction networks. (a) Direct and (b) dual-rail CRNs computing the function $f(x_1, x_2) = \max(x_1, x_2)$ plotted in (c). (d) Dual-rail CRN computing the function $f(x_1, x_2) = x_1 + \min(-x_1, x_2) - \max(x_1, x_2)$ plotted in (e). (f) Direct CRN computing the (discontinuous but still positive-continuous; see Definition 5.8) function $f(x_1, x_2) = x_1$ if $x_2 = 0$ and $2x_1$ if $x_2 > 0$ plotted in (g).

concentrations of C, X, Y , respectively, by $0, -0.5, +1.5$.) Subject to this constraint, the widest class of trajectories that still satisfies the intuitive meaning of the reaction semantics can be described informally as follows: (1) concentrations cannot become negative; (2) all reactants must be present when a reaction occurs (e.g., if a reaction uses a catalyst,² then the catalyst must be present); (3) the causal relationships between the production of species is respected (e.g., if producing A requires B and producing B requires A , then neither can ever be produced if both are absent).³ This notion of “allowed trajectories” is formalized as Definition 2.22.

The example shown in Figure 1(a) illustrates the style of computation studied here. Let $f : \mathbb{R}_{\geq 0}^2 \rightarrow \mathbb{R}_{\geq 0}$ be the max function $f(x_1, x_2) = \max(x_1, x_2)$ restricted to non-negative x_1 and x_2 . The CRN of Figure 1(a) computes this function in the following sense: Inputs x_1 and x_2 are given as initial concentrations of input species X_1 and X_2 . Then the CRN converges to f ’s output value $\max(x_1, x_2)$ of species Y , under a very wide interpretation of rate laws. Intuitively, the first two reactions must eventually produce $x_1 + x_2$ of Y , and x_1, x_2 of Z_1 and Z_2 , respectively. This is enforced by the stoichiometric constraint that the amount of Z_1 and Y produced is equal to the amount of X_1 consumed (and analogously for the second reaction). Stoichiometric constraints require the third reaction to produce the amount of K that is the minimum of the amount of Z_1 and Z_2 eventually produced in the first two reactions. Thus, $\min(x_1, x_2)$ of K is eventually produced. Therefore, the fourth reaction eventually consumes $\min(x_1, x_2)$ molecules of Y leaving $x_1 + x_2 - \min(x_1, x_2) =$

²A species acts catalytically in a reaction if it is both a reactant and product: e.g., C in reaction $A + C \rightarrow B + C$. Note that executing this reaction without C does not by itself violate condition (1).

³See Section 2.4 for examples showing that in the continuous setting conditions (2) and (3) are not mutually redundant.

$\max(x_1, x_2)$ of Y behind. We can imagine an adversary pushing flux through these four reactions in any devious stratagem (i.e., arbitrary rates), yet unable to prevent the CRN from converging to the correct output, so long as applicable reactions must eventually occur.

We further consider the natural extension of such computation to handle negative real values. The example shown in Figure 1(b) computes $f(x_1, x_2) = \max(x_1, x_2)$ ($f : \mathbb{R}^2 \rightarrow \mathbb{R}$), graphed in (c). To handle negative input and output values, we represent the value of each input and output by a pair of species using the so-called “dual-rail” representation. For example, in state c , $x_1 = c(X_1^+) - c(X_1^-)$ —i.e., the difference between the concentrations of species X_1^+ and X_1^- . Note that when X_1^- and X_2^- are initially absent, the CRN becomes equivalent to the first three reactions of Figure 1(a) under relabeling of species. We do not need the last reaction of (a), because the output is represented as the difference of Y^+ and Y^- by our convention. For the argument that the computation is correct even if X_1^- and X_2^- are initially present, we refer the reader to the proof of Corollary 5.14 in Section 5.3.

In addition to handling negative values, the dual-rail representation has the benefit of allowing composition. Specifically, the dual-rail representation allows a CRN to never consume its output species (e.g., rather than consuming Y^+ , it can produce Y^-). This monotonicity in the production of output allows directly composing CRN computations simply by concatenating CRNs and relabeling species (e.g., to make the output of one be input to the other). Since the upstream CRN never consumes its output species, the downstream CRN is free to consume them without interfering with the upstream computation. Since the class of functions computable by dual-rail CRNs ends up being invariant to whether or not, they are allowed to consume their output, and our results imply that dual-rail computation is composable without sacrificing computational power (see Section 5.1).

1.1 Summary of Main Results

Our first contribution is to define a reachability relation that captures the broadest reasonable notion of “what could happen” and is of independent interest. Although concentration trajectories of mass-action kinetics (and other standard rate laws) follow smooth curves, we base the reachability relation on taking simple-to-analyze straight-line paths. Theorem 2.27 shows that this notion of reachability is exactly equivalent to satisfying the three intuitive properties described above, (1) nonnegative concentrations, (2) reactions require their reactants present, and (3) respecting causal relationships between the production of species. Thus, our reachability relation has all reasonable rate laws as special cases; i.e., if any of them can reach a state, then so can our reachability relation.

The reachability relation allows us to formally define *stable computation* in Definition 3.2, analogously to similar definitions of probability 1 computation in discrete systems [6, 18]. Stable computation allows us to delineate when a function *cannot* be computed rate-independently. Roughly, unless the system is stably computing, then an adversary can always push it “far” away from the correct output (Theorem 3.3), precluding the system from being reasonably rate-independent.

For the positive direction, the CRN should converge to the correct output no matter the reaction rates. While a CRN that does not stably compute is not rate-independent (which is sufficient for negative results), the positive direction does not directly follow from stable computation for continuous systems. Indeed, we show examples of CRNs that stably compute a function by our definition, yet under standard mass-action kinetics fail to converge to the correct output; see Section 4. Instead, we capture a very strong notion of “convergence despite perturbations” in *fair computation* (Definition 4.3), based on generalized rate laws (so-called fair rate schedules, Definition 4.1). A CRN that fairly computes converges to the correct output as time $t \rightarrow \infty$ under any trajectory satisfying the three intuitive conditions above, plus an additional requirement that reactions *do*

occur when applicable. (In particular, mass-action (Corollary 4.11) satisfies these conditions, but the range of rate laws satisfying the conditions is much broader.) Luckily, stable computation and fair computation can be tightly connected, and we show that for a special class of CRNs we call *feedforward* (Definition 4.5) the two definitions coincide. In other words, a feedforward CRN stably computes a function if and only if it fairly computes the function (Lemmas 4.9 and 4.4). We show that all functions stably computable by CRNs are computable by feedforward CRNs (Lemmas 5.16 and 5.15), implying that the class of functions computable by CRNs under either definition—stable computation or fair computation—is identical. In other words, we can freely work with the simpler definition of stable computation, knowing that we are actually reasoning about a very general notion of rate-independence.

The above line of reasoning leads us to conclude that exactly the functions that are positive-continuous, piecewise linear (direct) or continuous, piecewise linear (dual-rail) can be rate-independently computed (Theorems 5.9 and 5.10). Positive-continuous means that the only discontinuities occur on a “face” of $\mathbb{R}_{\geq 0}^k$ —i.e., the function may discontinuously jump only at a point where some input goes from 0 to positive. We already saw a simple example of a continuous, piecewise linear function (max function, Figure 1(a,b,c)). Figure 1(d,e) shows a more complex example and a CRN that computes it. Figure 1(f,g) shows a discontinuous but positive-continuous function and a CRN that computes it. Although our work shows that the computational power of rate-independent CRNs is limited, the power of the computable class of functions should not be underestimated. For example, allowing a fixed non-zero initial concentration of non-input species (see Section 6.2), such CRNs are equivalent to ReLU neural networks—arguably the most widely used type of neural networks in machine learning [52].

1.2 Chemical Motivation

Traditionally, CRNs have been used as a descriptive language to analyze naturally occurring chemical reactions, as well as various other systems with a large number of interacting components such as gene regulatory networks and animal populations. However, CRNs also constitute a natural choice of programming language for engineering artificial systems. For example, nucleic-acid networks can be rationally designed to implement arbitrary chemical reaction networks [14, 20, 50, 51]. Thus, since in principle any CRN can be physically built, hypothetical CRNs with interesting behaviors are becoming of more than theoretical interest. One day artificial CRNs may underlie embedded controllers for biochemical, nanotechnological, or medical applications, where environments are inherently incompatible with traditional electronic controllers. However, to effectively program chemistry, we must understand the computational power at our disposal. In turn, the computer science approach to CRNs is also beginning to generate novel insights regarding natural cellular regulatory networks [15].

At the fine-grained level of detail, chemistry is discrete and stochastic. This level is typically modeled by discrete CRNs, where the state is a vector of nonnegative *integers* representing the counts of each species in the given reaction vessel, and reactions are modeled by a Markov jump process [29]. The continuous model is governed by a system of mass-action ordinary differential equations, which can be derived as a limiting case of the discrete model when volume and counts are large [34].⁴ While of physical primacy, the discrete model can be less suitable for reasoning

⁴The exact statement of Kurtz’s convergence result [34] is beyond the scope of this article. It considers taking a discrete CRN with initial integer molecular counts given by vector $\mathbf{c} \in \mathbb{N}^k$ in volume $V > 0$, then “scaling up” by factor $n \in \mathbb{N}$, i.e., considering the discrete CRN with initial state $n \cdot \mathbf{c}$ in volume $n \cdot V$. The result, stated very roughly, is that with high probability the n -scaled CRN has a trajectory (dividing discrete counts by $n \cdot V$ to convert to units of concentration) that stays close to the real-valued mass-action concentration trajectory, but only for time $O(\log n)$. An example CRN where this time bound is tight is $A + B \rightarrow A + B + Y$, $A \rightarrow 2A$, $B \rightarrow \emptyset$, starting with 1A, 1B. In mass-action, the concentrations

about feasible chemical algorithms. Many algorithms in the discrete model rely on a single molecule (called a “leader”) to coordinate computation [8]. Whether the initial state is assumed to have a leader or the CRN is designed to eliminate all but one copy of the leader species (“leader election”), such algorithms relying on single-molecule behavior are currently infeasible, since any single molecule can get damaged or become effectively lost.

An important reason for our focus on *stoichiometric computation* is that algorithms relying only on stoichiometry make easier design targets. The rates of reactions are real-valued quantities that can fluctuate with reaction conditions such as temperature, while the stoichiometric coefficients are immutable whole numbers set by the nature of the reaction. Methods for physically implementing CRNs naturally yield systems with digital stoichiometry that can be set exactly [14, 50], whereas these methods often suffer from imprecise control over reaction rates [20, 51]. Further, relying on specific rate laws can be problematic: Many systems do not apparently follow mass-action rate laws, and chemists have developed an array of alternative rate laws such as Michaelis-Menten (modeling enzymes) and Hill-function kinetics (widely used for gene regulation).⁵ It is well-known that cells are not well-mixed, and many models have been developed to take space into account (e.g., reaction-diffusion [32]). Moreover, robustness of rate laws is a recurring motif in systems biology due to much evidence that biological regulatory networks tend to be robust to the form of the rate laws and the rate parameters [10]. Thus, we are interested in what computations can be understood or engineered without regard for the reaction rate laws.

1.3 Related Works

An earlier conference version of this article appeared as Reference [19]. Besides replacing a number of informal arguments with rigorous proofs, this journal version also expands and generalizes the results of the conference version. For example, we introduce new machinery for representing and manipulating trajectories as linear objects (piecewise linear paths). We also define a broad class of rate laws, formalized by Definition 2.22, which captures mass-action kinetics and all other known rate laws such as Michaelis-Menten and Hill-function kinetics, and we prove that our definition of reachability is as general as any in this class. For the constructive part, this version also generalizes Lemma 3.4 of Reference [19] (in addition to correcting its proof) by introducing feedforward CRNs and proving that correct computation in our setting implies convergence under any “reasonable” rate law (one that produces a *fair* schedule of rates; Definition 4.1) for any feedforward CRN (Lemma 4.8).

The relationship between the discrete and continuous CRN models is a complex and much studied one in the natural sciences [45]. The computational abilities of discrete CRNs have been investigated more thoroughly than of continuous CRNs and have been shown to have a surprisingly rich computational structure. Of most relevance here is the work in the discrete setting showing that the class of functions that can be computed depends strongly on whether the computation must be correct or just likely to be correct (under the usual stochastic kinetics)—which is the discrete version of the distinction between rate-independent and rate-dependent computation. While Turing universal computation is possible with an arbitrarily small, non-zero probability of error over all time [49], forbidding error altogether limits the computational power: Error-free computation by stochastic CRNs is limited to semilinear predicates and functions [6, 18]. (Intuitively, semilinear

of A and B at time t are, respectively, e^t and e^{-t} , whose product is the constant 1, so the first reaction produces Y at a unit rate forever. However, scaling up to nA , nB , the discrete CRN consumes all B in $O(\log n)$ time, at which point production of Y halts. See References [24, 35] for other example CRNs for which the two models diverge after sufficient time.

⁵It is generally supposed that chemical reactions would follow mass-action if properly decomposed into truly elementary reactions and the solution is well-mixed. For example, Michaelis-Menten and Hill-function kinetics can be derived as a limiting case of mass-action when the reaction is initiated and completed at vastly different time scales.

functions are expressible as a finite union of affine functions, with “simple, periodic” domains of each affine function [18].) The study of error-free computation in discrete CRNs is heavily based on the results first developed for a model of distributed computing called population protocols [6, 9]. We formally refer to our notion of rate-independent computation as *stable computation* in direct reference to the analogous notion in population protocols.

While our notion of rate-independent computation is the natural extension of deterministic computation in the discrete model, there are many differences between the two settings. As mentioned above, many discrete algorithms such as those that rely on a single “leader” molecule fail to work in the continuous setting, and some functions like distinguishing between even and odd molecular counts do not make sense. Broadly speaking, the proof techniques appear to require very different machinery, and the importance of stable computation itself needs substantial justification in the continuous model (as the examples shown at the beginning of Section 4 demonstrate).

Continuous CRNs have been proven to be Turing universal under mass-action rate laws [27], a consequence of the surprising computational power of polynomial ODEs [12]. In ODEs without the CRN semantics, there is no natural notion of stoichiometry and thus no notion of rate-independence analogous to ours. In chemistry, the same physical process (a reaction) is responsible for multiple monomials across multiple ODEs, which justifies these monomials being exactly the same or in fixed ratios (corresponding to obeying reaction stoichiometry). Such forced relationships do not seem natural for more general polynomial ODEs that do not correspond to chemical reactions.⁶

Our notion of reachability (Definition 2.3) is intended to capture a wide diversity of possible rate laws. Generalized rate laws (extending mass-action, Michaelis-Menten, etc.) have been previously studied, although not in a computational setting. For example, certain conditions were identified on global convergence to equilibrium based on properties intuitively similar to ours [4]. A related idea in the literature, generalizing mass-action, is differential inclusion [30]. In that model, the mass-action rate constants are not fixed to be particular real numbers constant over time, but instead can vary over time within some bounded interval $[l, u]$ fixed in advance, with $0 < l \leq u < \infty$. Another related idea is the notion of a reaction system [26], which generalizes even beyond mass-action, allowing reaction rates to be an (almost) arbitrary function of species concentrations.⁷ Other generalized rate laws have been defined as well [5, 23].

Since the original publication of the conference version of this article [19], a number of works have used our framework. A key concept in capturing rate-independent computation is the reachability relation (segment-reachability, Definition 2.3). Reference [16] showed that, given two states, deciding whether one is reachable from the other is solvable in polynomial time. This contrasts sharply with the hardness of the reachability problem for discrete CRNs, which, although computable [39], is not even primitive recursive [22, 37]. (These results were proven using the terminology of the equivalent models of Petri nets/vector addition systems.)

The question of deciding whether a given CRN is rate-independent was studied in Reference [23]. The work provides sufficient graphical conditions on the structure of the CRN that ensure rate-independence for the whole CRN or only for certain output species. Interestingly, the authors

⁶For example, consider the reaction $A \rightarrow 2B$, with ODEs $\dot{a} = -a$ and $\dot{b} = 2a$. One can imagine a “chemical” adversary adjusting the rate of the reaction to speed it up or slow it down, but what the adversary *cannot* control is that to consume x amount of A requires producing exactly $2x$ amount of B and vice versa. This connection between the rates of consumption of A and production of B does not have an obvious counterpart in more general polynomial ODEs and analog computational models.

⁷Our notion of valid rate schedules in Definition 2.22 is even more general than a reaction system in that a valid rate schedule does not require a reaction’s rate to be a function of species concentrations, for instance, allowing an adversary to visit the same state twice but apply different reaction rates each time.

of Reference [23] applied this method to the Biomodels repository of curated CRNs of biological origin and found a number of CRNs that satisfy the rate-independence conditions.

An important motivation for the dual-rail representation in this work is to allow composition of rate-independent CRN modules (Section 5.1). Such rate-independent modules can be composed into overall rate-independent computation simply by concatenating their chemical reactions and relabeling species (such that the output species of the first is the input species of the second, and all other species are distinct). In contrast, rate-independent composition with the direct (non-dual rail) representation, introduces an additional “superadditivity” constraint that for all input vectors \mathbf{x} and \mathbf{x}' , $f(\mathbf{x}) + f(\mathbf{x}') \leq f(\mathbf{x} + \mathbf{x}')$ [17]. Thus, for example, the non-superadditive max function (Figure 1) provably cannot be composably computed with a rate-independent CRN in the direct representation. Composable computation has also been characterized in the discrete model [31, 48].

Other input encodings have been considered besides direct and dual-rail. For example, the so-called “fractional encoding” encodes a real number between 0 and 1 as a ratio $\frac{x_1}{x_0 + x_1}$ where x_0, x_1 are concentrations of two input species [44]. Other notions of chemical “rate-independence” include CRNs that work independently of the rate law as long as there is a separation into fast and slow reactions [47]. For a detailed survey on computation with CRNs (both continuous and discrete), see Reference [13].

2 DEFINING REACHABILITY IN CHEMICAL REACTION NETWORKS

2.1 Chemical Reaction Networks

We first explain our notation for vectors of concentrations of chemical species and then formally define chemical reaction networks.

Given a finite set F , let \mathbb{R}^F denote the set of functions $\mathbf{c} : F \rightarrow \mathbb{R}$. We view \mathbf{c} equivalently as a vector of real numbers indexed by elements of F . Given $x \in F$, we write $\mathbf{c}(x)$, or sometimes c_x , to denote the real number indexed by x . The notation $\mathbb{R}_{\geq 0}^F$ is defined similarly for nonnegative real vectors. Throughout this article, let Λ be a finite set of chemical *species*. Given $S \in \Lambda$ and $\mathbf{c} \in \mathbb{R}_{\geq 0}^\Lambda$, we refer to $\mathbf{c}(S)$ as the *concentration of S in \mathbf{c}* . For any $\mathbf{c} \in \mathbb{R}_{\geq 0}^\Lambda$, let $[\mathbf{c}] = \{S \in \Lambda \mid \mathbf{c}(S) > 0\}$, the set of species *present* in \mathbf{c} (a.k.a., the *support* of \mathbf{c}). We write $\mathbf{c} \leq \mathbf{c}'$ to denote that $\mathbf{c}(S) \leq \mathbf{c}'(S)$ for all $S \in \Lambda$. Given $\mathbf{c}, \mathbf{c}' \in \mathbb{R}_{\geq 0}^\Lambda$, we define the vector component-wise operations of addition $\mathbf{c} + \mathbf{c}'$, subtraction $\mathbf{c} - \mathbf{c}'$, and scalar multiplication $x\mathbf{c}$ for $x \in \mathbb{R}$. If $\Delta \subset \Lambda$, we view a vector $\mathbf{c} \in \mathbb{R}_{\geq 0}^\Lambda$ equivalently as a vector $\mathbf{c} \in \mathbb{R}_{\geq 0}^\Delta$ by assuming $\mathbf{c}(S) = 0$ for all $S \in \Lambda \setminus \Delta$. For $\Delta \subset \Lambda$, we write $\mathbf{c} \upharpoonright \Delta$ to denote \mathbf{c} *restricted to Δ* ; in particular, $\mathbf{c} \upharpoonright \Delta = \mathbf{0} \iff (\forall S \in \Delta) \mathbf{c}(S) = 0$. (We use the convention that $\mathbf{c} \upharpoonright \emptyset = \mathbf{0}$ for all states \mathbf{c} .)

A *reaction* over Λ is a pair $\alpha = \langle \mathbf{r}, \mathbf{p} \rangle \in \mathbb{N}^\Lambda \times \mathbb{N}^\Lambda$, such that $\mathbf{r} \neq \mathbf{p}$, specifying the stoichiometry of the reactants and products, respectively.⁸ For instance, given $\Lambda = \{A, B, C\}$, the reaction $A + 2B \rightarrow A + 3C$ is the pair $\langle (1, 2, 0), (1, 0, 3) \rangle$. We represent reversible reactions such as $A \rightleftharpoons B$ as two irreversible reactions $A \rightarrow B$ and $B \rightarrow A$. In this article, we assume that $\mathbf{r} \neq \mathbf{0}$, i.e., we have no reactions of the form $\emptyset \rightarrow \dots$.⁹ A (finite) **chemical reaction network (CRN)** is a pair $C = (\Lambda, R)$, where Λ is a finite set of chemical *species*, and R is a finite set of reactions over Λ . A *state* of a CRN $C = (\Lambda, R)$ is a vector $\mathbf{c} \in \mathbb{R}_{\geq 0}^\Lambda$. Given a state \mathbf{c} and reaction $\alpha = \langle \mathbf{r}, \mathbf{p} \rangle$, we say that α is *applicable* in \mathbf{c} if $[\mathbf{r}] \subseteq [\mathbf{c}]$ (i.e., \mathbf{c} contains positive concentration of all of the reactants). If no

⁸It is customary to define, for each reaction, a *rate constant* $k \in \mathbb{R}_{>0}$ specifying a constant multiplier on the mass-action rate (i.e., the product of the reactant concentrations), but as we are studying CRNs whose output is independent of the reaction rates, we leave the rate constants out of the definition.

⁹We allow high-order reactions; i.e., those that have more than two reactants. Such higher order reactions could be eliminated from our constructions using the transformation that replaces $S_1 + S_2 + \dots + S_n \rightarrow P_1 + \dots + P_m$ with bimolecular reactions $S_1 + S_2 \rightleftharpoons S_{12}$, $S_{12} + S_3 \rightleftharpoons S_{123}$, $S_{123} + S_4 \rightleftharpoons S_{1234}$, \dots , $S_n + S_{12\dots n-1} \rightarrow P_1 + \dots + P_m$.

reaction is applicable in state \mathbf{c} , then we say \mathbf{c} is *static*. We say a species S is *produced* in reaction $\langle \mathbf{r}, \mathbf{p} \rangle$ if $\mathbf{r}(S) < \mathbf{p}(S)$, and *consumed* if $\mathbf{r}(S) > \mathbf{p}(S)$. (Note that a catalyst, such as C in the reaction $C + X \rightarrow C + Y$, is neither produced nor consumed.)

2.2 Segment Reachability

In the previous section, we defined the syntax of CRNs. Toward studying rate-independent computation, we now want to define the semantics of what “could happen” if reaction rates can vary arbitrarily over time. This is captured by a notion of *reachability*, which is the focus of this section. Intuitively, \mathbf{d} is reachable from \mathbf{c} if applying some amount of reactions to \mathbf{c} results in \mathbf{d} , such that no reaction is ever applied when any of its reactants are concentration 0. Formalizing this concept is a bit tricky and constitutes one of the contributions of this article. Intuitively, we will think of reachability via straight line segments. This may appear overly limiting; after all mass-action and other rate laws trace out smooth curves. However, in this and subsequent sections, we show a number of properties of our definition that support its reasonableness.

Throughout this section, fix a CRN $C = (\Lambda, R)$. All states \mathbf{c} , and so on, are assumed to be states of C . We define the $|\Lambda| \times |R|$ *reaction stoichiometry matrix* \mathbf{M} such that, for species $S \in \Lambda$ and reaction $\alpha = \langle \mathbf{r}, \mathbf{p} \rangle \in R$, $\mathbf{M}(S, \alpha) = \mathbf{p}(S) - \mathbf{r}(S)$ is the net amount of S produced by α (negative if S is consumed).¹⁰ For example, if we have the reactions $X \rightarrow Y$ and $X + A \rightarrow 2X + 3Y$, and if the three rows correspond to X , A , and Y , in that order, then

$$\mathbf{M} = \begin{pmatrix} -1 & 1 \\ 0 & -1 \\ 1 & 3 \end{pmatrix}.$$

Definition 2.1. State \mathbf{d} is *straight-line reachable* (a.k.a. *1-segment reachable*) from state \mathbf{c} , written $\mathbf{c} \rightarrow^1 \mathbf{d}$, if $(\exists \mathbf{u} \in \mathbb{R}_{\geq 0}^R) \mathbf{c} + \mathbf{M}\mathbf{u} = \mathbf{d}$ and $\mathbf{u}(\alpha) > 0$ only if reaction α is applicable at \mathbf{c} . In this case, write $\mathbf{c} \rightarrow_{\mathbf{u}}^1 \mathbf{d}$.

Intuitively, by a single segment, we mean running the reactions applicable at \mathbf{c} at a constant (possibly 0) rate to get from \mathbf{c} to \mathbf{d} . In the definition, $\mathbf{u}(\alpha)$ represents the flux of reaction $\alpha \in R$.

The next definition is used in our main notion of reachability, which uses either a finite number of straight lines or infinitely many so long as they converge to a single state.

Definition 2.2. Let $k \in \mathbb{N} \cup \{\infty\}$. State \mathbf{d} is *k-segment reachable* from state \mathbf{c} , written $\mathbf{c} \rightsquigarrow^k \mathbf{d}$, if $(\exists \mathbf{b}_0, \dots, \mathbf{b}_k) \mathbf{c} = \mathbf{b}_0 \rightarrow^1 \mathbf{b}_1 \rightarrow^1 \mathbf{b}_2 \rightarrow^1 \dots \rightarrow^1 \mathbf{b}_k$, with $\mathbf{b}_k = \mathbf{d}$ if $k \in \mathbb{N}$, or $\lim_{i \rightarrow \infty} \mathbf{b}_i = \mathbf{d}$ if $k = \infty$.

Definition 2.3. State \mathbf{d} is *segment-reachable* (or simply *reachable*) from state \mathbf{c} , written $\mathbf{c} \rightsquigarrow \mathbf{d}$, if $(\exists k \in \mathbb{N} \cup \{\infty\}) \mathbf{c} \rightsquigarrow^k \mathbf{d}$.

For example, suppose the reactions are $X \rightarrow C$ and $C + Y \rightarrow C + Z$, and we are in state $\{0C, 1X, 1Y, 0Z\}$. With straight-line segments, any state with a positive amount of Z must be reached in at least two segments: first to produce C , which allows the second reaction to occur, and then any combination of the first and second reactions. For example, $\{0C, 1X, 1Y, 0Z\} \rightarrow^1 \{0.1C, 0.9X, 1Y, 0Z\} \rightarrow^1 \{1C, 0X, 0Y, 1Z\}$. This is a simple example showing that more states are reachable with \rightsquigarrow than \rightarrow^1 . Often, Definition 2.3 is used implicitly, when we make statements such as, “Run reaction 1 until X is gone, then run reaction 2 until Y is gone,” which implicitly defines two straight lines in concentration space.

Although more effort will be needed to justify its reasonableness (see Section 2.4), segment-reachability will serve as the main notion of reachability in this article.

¹⁰Note that \mathbf{M} does not fully specify C , since catalysts are not modeled: reactions $Z + X \rightarrow Z + Y$ and $X \rightarrow Y$ both correspond to the column vector $(-1, 1, 0)^\top$.

2.3 Bound on Number of Required Line Segments in Segment Reachability

It may seem that we can never achieve the “full diversity” of states reachable with an infinite number of line segments if we use only a bounded number of line segments. However, Theorem 2.15 shows that increasing the number of straight-line segments beyond a certain point does not make any additional states reachable. Thus, using a few line segments captures all the states reachable with arbitrarily many line segments, and in fact even in the limit of infinitely many line segments.

To prove Theorem 2.15, we first develop important machinery for representing and manipulating paths under \rightsquigarrow . Note that reachability is closed under addition and scaling in the sense that if $\mathbf{c} \rightsquigarrow \mathbf{d}$ and $\mathbf{c}' \rightsquigarrow \mathbf{d}'$, then $\alpha\mathbf{c} + \beta\mathbf{c}' \rightsquigarrow \alpha\mathbf{d} + \beta\mathbf{d}'$ for all $\alpha, \beta \in \mathbb{R}_{\geq 0}$. The following definition captures this property by defining a linear space of all paths. This machinery will also be key to proving the piecewise linearity of the computed function in Section 5.5.

Definition 2.4. Let $C = (\Lambda, R)$ be a CRN with species Λ and reactions R . For $n \in \mathbb{N}$, we define a linear map $\mathbf{x}_n : (\mathbb{R}^\Lambda \times \times_{i=1}^\infty \mathbb{R}^R) \rightarrow \mathbb{R}^\Lambda$, which takes $\boldsymbol{\gamma} = (\mathbf{x}_0, \mathbf{u}_1, \mathbf{u}_2, \dots)$ representing an initial state \mathbf{x}_0 and reaction flux vectors $\mathbf{u}_1, \mathbf{u}_2, \dots$, and produces

$$\mathbf{x}_n(\boldsymbol{\gamma}) = \mathbf{x}_0 + \sum_{i=1}^n \mathbf{M}\mathbf{u}_i,$$

which intuitively is the state reached after traversing the first n line segments. Let Ψ be the set of $\boldsymbol{\gamma}$ for which $\lim_{n \rightarrow \infty} \mathbf{x}_n(\boldsymbol{\gamma})$ converges. We call elements of Ψ *prepaths*.

Definition 2.4 allows a prepath to be essentially any sequence of vectors in the linear subspace spanned by reaction vectors. The next definition restricts the vectors with three physical constraints: Species concentrations are nonnegative, reaction fluxes are nonnegative (i.e., reactions can only go one way, turning reactants into products), and reactions cannot occur if any reactant is 0.

Definition 2.5. Let Γ_∞ be the subset of Ψ consisting of vectors $\boldsymbol{\gamma} = (\mathbf{x}_0, \mathbf{u}_1, \mathbf{u}_2, \dots)$ satisfying the following conditions for all $n \in \mathbb{N}$:

- (1) $\mathbf{x}_n(\boldsymbol{\gamma}) \in \mathbb{R}_{\geq 0}^\Lambda$.
- (2) $\mathbf{u}_n \in \mathbb{R}_{\geq 0}^R$.
- (3) every reaction with positive flux in \mathbf{u}_{n+1} is applicable at \mathbf{x}_n .

We call an element of Γ_∞ a *piecewise linear path* or sometimes just a *path*.

Definitions 2.4 and 2.5 allow an infinite sequence of reaction flux vectors (each corresponding to a straight line in the definition of 1-segment reachability). A finite number of straight lines can be specified by letting all but finitely many $\mathbf{u}_i = \mathbf{0}$. The next definition bounds how many can be nonzero.

Definition 2.6. For $k \in \mathbb{N}$, define Γ_k to be the subset of Γ_∞ consisting of all paths $\boldsymbol{\gamma} = (\mathbf{x}_0, \mathbf{u}_1, \mathbf{u}_2, \dots)$ such that $\mathbf{u}_i = \mathbf{0}$ for all $i > k$. Say that a path is *finite* if it is contained in Γ_k for some $k \in \mathbb{N}$.

Intuitively, Γ_∞ is the space of all of the valid piecewise linear paths that the system can take starting from any given initial state and Γ_k ($k \in \mathbb{N}$) is the set of all such paths that have length at most k ; thus, $\Gamma_0 \subseteq \Gamma_1 \subseteq \dots \subseteq \Gamma_\infty$.

LEMMA 2.7. For $k \in \mathbb{N} \cup \{\infty\}$, Γ_k is convex.

PROOF. Let $\boldsymbol{\gamma}_0, \boldsymbol{\gamma}_1 \in \Gamma_\infty$ be two paths and consider $\lambda \in (0, 1)$. We need to show that $\boldsymbol{\gamma}_\lambda = (1 - \lambda)\boldsymbol{\gamma}_0 + \lambda\boldsymbol{\gamma}_1$ is in Γ_∞ . Recall that $\mathbf{x}_n(\boldsymbol{\gamma})$ is the state reached after the first n segments of path $\boldsymbol{\gamma}$. Note that for any $n \in \mathbb{N}$,

$$\mathbf{x}_n(\boldsymbol{\gamma}_\lambda) = (1 - \lambda)\mathbf{x}_n(\boldsymbol{\gamma}_0) + \lambda\mathbf{x}_n(\boldsymbol{\gamma}_1).$$

Since $\mathbb{R}_{\geq 0}^\Lambda$ is convex and both $\mathbf{x}_n(\gamma_1)$ and $\mathbf{x}_n(\gamma_2)$ are in $\mathbb{R}_{\geq 0}^\Lambda$, we conclude that $\mathbf{x}_n(\gamma_\lambda)$ is in $\mathbb{R}_{\geq 0}^\Lambda$, too. Moreover, because both $\lim_{n \rightarrow \infty} \mathbf{x}_n(\gamma_0)$ and $\lim_{n \rightarrow \infty} \mathbf{x}_n(\gamma_1)$ converge, we see that

$$\lim_{n \rightarrow \infty} \mathbf{x}_n(\gamma_\lambda) = (1 - \lambda) \lim_{n \rightarrow \infty} \mathbf{x}_n(\gamma_0) + \lambda \lim_{n \rightarrow \infty} \mathbf{x}_n(\gamma_1)$$

also converges.

Below, for a path $\gamma = (\mathbf{x}_0, \mathbf{u}_1, \mathbf{u}_2, \dots)$, we use the notation $\mathbf{u}_n(\gamma)$ to represent the n th flux vector \mathbf{u}_n in γ . Since

$$\mathbf{u}_n(\gamma_\lambda) = (1 - \lambda)\mathbf{u}_n(\gamma_0) + \lambda\mathbf{u}_n(\gamma_1),$$

any reaction α occurs with positive flux in $\mathbf{u}_n(\gamma_\lambda)$ only if α occurs with positive flux in $\mathbf{u}_n(\gamma_i)$ for $i = 0$ or 1 . Without loss of generality, suppose that α occurs with positive flux in $\mathbf{u}_n(\gamma_0)$. Then reaction α is applicable at $\mathbf{x}_{n-1}(\gamma_0)$, so the reactants are all present in positive concentrations in $\mathbf{x}_{n-1}(\gamma_0)$. This implies that they are present with positive concentrations in $\mathbf{x}_{n-1}(\gamma_\lambda)$ (note that we have excluded the case $\lambda = 1$ from the outset). Therefore, reaction α is applicable at $\mathbf{x}_{n-1}(\gamma_\lambda)$. We conclude that every reaction occurring with positive flux in $\mathbf{u}_n(\gamma_\lambda)$ is applicable at $\mathbf{x}_{n-1}(\gamma_\lambda)$. This shows that Γ_∞ is convex.

To see that Γ_k for $k \in \mathbb{N}$ is also convex, note that if $\mathbf{u}_n(\gamma_0) = 0$ and $\mathbf{u}_n(\gamma_1) = 0$, then $\mathbf{u}_n(\gamma_\lambda)$ will also be zero. \square

The next lemma shows that if it is possible to reach from a state \mathbf{c} to several other states, each containing some species possibly distinct from each other, then it is possible to reach from \mathbf{c} to a state with *all* of those species present at once.

LEMMA 2.8. *Let $l \in \mathbb{N}$, $k \in \mathbb{N} \cup \{\infty\}$, and let $\mathbf{c}, \mathbf{d}_1, \dots, \mathbf{d}_l$ be states such that $\mathbf{c} \rightsquigarrow^k \mathbf{d}_1$, $\mathbf{c} \rightsquigarrow^k \mathbf{d}_2$, \dots , and $\mathbf{c} \rightsquigarrow^k \mathbf{d}_l$. Then there exists \mathbf{d} such that $\mathbf{c} \rightsquigarrow^k \mathbf{d}$ and $[\mathbf{d}] = \bigcup_{i=1}^l [\mathbf{d}_i]$.*

PROOF. Write γ_i for the path from \mathbf{c} to \mathbf{d}_i ; the convexity of Γ_k shows that the convex combination

$$\gamma = \frac{1}{l} \sum_{i=1}^l \gamma_i$$

is a valid path in Γ_k . Letting

$$\mathbf{d} = \frac{1}{l} \sum_{i=1}^l \mathbf{d}_i,$$

γ exhibits a k -segment path from \mathbf{c} to \mathbf{d} . If S is a species that is present at \mathbf{d}_i for any i , then S is also present at \mathbf{d} . However, if S is present in none of the \mathbf{d}_i , then S is not present in \mathbf{d} . As a result, $[\mathbf{d}] = \bigcup_{i=1}^l [\mathbf{d}_i]$. \square

Definition 2.9. Given a state \mathbf{c} , let $P(\mathbf{c})$ be the set of all species that are *producible* from \mathbf{c} —i.e., present in some state that is segment-reachable from \mathbf{c} .

The next lemma shows that with at most a constant number of straight line segments, it is possible to reach from any state \mathbf{c} to a state containing *all* species possible to produce from \mathbf{c} .

LEMMA 2.10. *Let m be the minimum of $|R|$ and $|\Lambda|$ and let \mathbf{c} be any state. Then there is a state \mathbf{d} such that $\mathbf{c} \rightsquigarrow^m \mathbf{d}$ and $[\mathbf{d}] = P(\mathbf{c})$.*

PROOF. Given a state \mathbf{c} , let $P_{<\infty}(\mathbf{c})$ be the set of all species that are present in some state that is k -segment-reachable from \mathbf{c} for some $k < \infty$.

We first argue that $P(\mathbf{c}) = P_{<\infty}(\mathbf{c})$. Since clearly $P_{<\infty}(\mathbf{c}) \subseteq P(\mathbf{c})$, it remains to show that $P(\mathbf{c}) \subseteq P_{<\infty}(\mathbf{c})$. Let S be a species that is present in the state \mathbf{d} such that $\mathbf{c} \rightsquigarrow^\infty \mathbf{d}$. Then there exists a

sequence $(b_i)_i$ such that $c \rightarrow^1 b_1 \rightarrow^1 b_2 \rightarrow^1 \dots$ with $d = \lim_{i \rightarrow \infty} b_i$. Because $S \in [d]$ there must be some $i_0 \in \mathbb{N}$ where $S \in [b_{i_0}]$, and, since $c \rightsquigarrow^{i_0} b_{i_0}$, we see that $S \in P_{<\infty}(c)$. Thus, $P(c) = P_{<\infty}(c)$.

We show that the lemma holds for $P_{<\infty}(c)$; since $P(c) = P_{<\infty}(c)$, this establishes the full lemma.

For all $i \in \mathbb{N}$, let Λ_i be the set of species S such that there exists a d with $c \rightsquigarrow^i d$ and $S \in [d]$. Similarly, let R_i be the set of reactions α such that there exists a d with $c \rightsquigarrow^i d$ and α is applicable in d . Note that $\Lambda_0 = [c]$ and R_0 is the set of reactions applicable in c . Also, since $c \rightsquigarrow^i d$ implies $c \rightsquigarrow^{i+1} d$, we see that $\Lambda_i \subseteq \Lambda_{i+1}$ and $R_i \subseteq R_{i+1}$ for all i .

Now, we show that for all i there exists some x_i such that $[x_i] = \Lambda_i$ and $c \rightsquigarrow^i x_i$ (and therefore R_i consists of the reactions applicable at x_i). To see this, for each $S \in \Lambda_i$ let d_S be a state such that $c \rightsquigarrow^i d_S$ and $S \in [d_S]$. By applying Lemma 2.8 to the set of all d_S , there is some d such that $c \rightsquigarrow^i d$ and $[d] = \Lambda_i$; this d is our desired x_i .

Now, we will show that if $\Lambda_i = \Lambda_{i+1}$, then $R_i = R_{i+1}$ and, independently, if $R_i = R_{i+1}$, then $\Lambda_{i+1} = \Lambda_{i+2}$ for all i . First suppose that $\Lambda_i = \Lambda_{i+1}$ and let α be a reaction in R_{i+1} . Then there is some state d such that $c \rightsquigarrow^{i+1} d$ and α is applicable at d . Since all of the reactants of α are present at d , they are a subset of $\Lambda_{i+1} = \Lambda_i$. They are therefore present at x_i , so α is applicable at x_i . We conclude that $\alpha \in R_i$ so $R_{i+1} = R_i$.

Now suppose that $R_i = R_{i+1}$ and let S be a species in Λ_{i+2} . Then there is some d such that $c \rightsquigarrow^{i+2} d$ and $S \in [d]$. If $S \in [c]$, then $S \in \Lambda_{i+1}$. Otherwise, S must be produced by some reaction α in $R_{i+1} = R_i$, and we can apply α to x_i to obtain a state d' such that $c \rightsquigarrow^i x_i \rightarrow^1 d'$ and $S \in [d']$. Again, we conclude that $S \in \Lambda_{i+1}$ so $\Lambda_{i+2} = \Lambda_{i+1}$.

Combining the two statements we just proved, we see that if $\Lambda_i = \Lambda_{i+1}$, then $\Lambda_i = \Lambda_j$ for all $j \geq i$, so $\Lambda_i = P_{<\infty}(c)$. Similarly, if $R_i = R_{i+1}$, then $\Lambda_{i+1} = P_{<\infty}(c)$.

If $|\Lambda| \leq |R|$, then, since $\Lambda_0 \subseteq \Lambda_1 \subseteq \dots$ is an increasing sequence of subsets of the finite set Λ , it must be the case that $\Lambda_j = \Lambda_{j+1}$ for some $j \leq |\Lambda|$, and in this case x_j gives our desired d . If, however, $|R| \leq |\Lambda|$, the proof is similar: First note that if $R_0 = \emptyset$, then we are done. Otherwise, $|R_0| \geq 1$ so, since R_i is an increasing sequence of subsets of R , there is some $j \leq |R| - 1$ such that $R_j = R_{j+1}$. Then $\Lambda_{j+1} = P_{<\infty}(c)$ so x_{j+1} gives our desired d . \square

Recall that a set is *closed* if it contains all of its limit points.

LEMMA 2.11. *Let $c \in \mathbb{R}_{\geq 0}^\Lambda$ be any state and let $S_c \subseteq \mathbb{R}_{\geq 0}^\Lambda$ be the set of states that are straight-line reachable from c . Then S_c is closed.*

PROOF. Let R_c be the set of reactions that are applicable at c . Then $C = \{u \in \mathbb{R}_{\geq 0}^R \mid u(\alpha) = 0 \text{ for } \alpha \notin R_c\}$ is a polyhedron. Then $c + MC$ is also a polyhedron (see Reference [53]), and is in particular closed. S_c is just $\mathbb{R}_{\geq 0}^\Lambda \cap (c + MC)$, and is therefore also closed. \square

Note that Lemma 2.11 is false if we replace “straight-line reachable” with “segment-reachable.” For example, consider $X \rightarrow C$ and $X + C \rightarrow Y + C$, where we take the initial state $c = \{1X, 0C, 0Y\}$. Note that for any $\varepsilon > 0$, we can reach the state $d_\varepsilon = \{0X, \varepsilon C, (1 - \varepsilon)Y\}$. However, because producing Y first requires consuming a positive amount of X to create the catalyst C , the state $d_0 = \{0X, 0C, 1Y\}$ is not segment reachable from c , even though $d_0 = \lim_{\varepsilon \rightarrow 0} d_\varepsilon$.

If we have an infinite sequence of states such that $c = b_0 \rightsquigarrow b_1 \rightsquigarrow b_2 \dots$ and $\lim_{i \rightarrow \infty} b_i = d$, then this does not immediately imply that $c \rightsquigarrow^\infty d$ by Definition 2.2. This is because although the endpoints of the paths $b_i \rightsquigarrow b_{i+1}$ converge to d , the intermediate states on the paths related by \rightarrow^1 (i.e., $b_i \rightarrow^1 b'_i \rightarrow^1 b''_i \rightarrow^1 \dots \rightarrow^1 b_{i+1}$) may not converge. To capture this weaker notion of convergence, we introduce the following definition, which generalizes \rightsquigarrow^∞ by requiring only that there be a converging *subsequence* of states. (The weaker notion will be eventually needed to prove Theorem 3.3.)

Definition 2.12. State \mathbf{d} is *s.s. segment reachable* from state \mathbf{c} , written $\mathbf{c} \rightsquigarrow_{ss}^{\infty} \mathbf{d}$, if $(\exists \mathbf{b}'_0, \mathbf{b}'_1, \dots) \mathbf{c} = \mathbf{b}'_0 \rightarrow^1 \mathbf{b}'_1 \rightarrow^1 \dots$, where for some subsequence $\mathbf{b}_0, \mathbf{b}_1, \dots$ of $\mathbf{b}'_0, \mathbf{b}'_1, \dots$, $\lim_{i \rightarrow \infty} \mathbf{b}_i = \mathbf{d}$.

The main results of this section have this weaker notion of convergence as a precondition, which will imply that, despite appearances, $\rightsquigarrow^{\infty}$ and $\rightsquigarrow_{ss}^{\infty}$ are actually equivalent.

The next lemma shows that if no more species are producible from state \mathbf{c} than are already present in \mathbf{c} , then any state \mathbf{d} that is $\rightsquigarrow_{ss}^{\infty}$ reachable from \mathbf{c} is reachable via a single straight line segment.

LEMMA 2.13. *If $[\mathbf{c}] = P(\mathbf{c})$, then every state \mathbf{d} such that $\mathbf{c} \rightsquigarrow_{ss}^{\infty} \mathbf{d}$ is straight-line reachable from \mathbf{c} (i.e., $\mathbf{c} \rightsquigarrow_{ss}^{\infty} \mathbf{d}$ implies $\mathbf{c} \rightarrow^1 \mathbf{d}$).*

PROOF. First consider the finite case where $\mathbf{c} \rightsquigarrow^k \mathbf{d}$ for $k < \infty$. Let $\mathbf{u}_1, \dots, \mathbf{u}_k$ be the flux vectors corresponding to the segments in the path from \mathbf{c} to \mathbf{d} . Then $\mathbf{u} = \mathbf{u}_1 + \dots + \mathbf{u}_k$ is a vector in $\mathbb{R}_{\geq 0}^R$. Every reaction that occurs with positive flux in \mathbf{u} has positive flux in one of the \mathbf{u}_i , and thus its reactants are in $P(\mathbf{c})$, so are present in \mathbf{c} by the assumption $[\mathbf{c}] = P(\mathbf{c})$. Thus, every reaction that occurs with positive flux in \mathbf{u} is applicable at \mathbf{c} . The straight-line from \mathbf{c} corresponding to \mathbf{u} takes \mathbf{c} to

$$\mathbf{c} + \mathbf{Mu} = \mathbf{c} + \sum_{i=1}^k \mathbf{Mu}_i = \mathbf{d},$$

so $\mathbf{c} \rightarrow_{\mathbf{u}}^1 \mathbf{d}$.

Now suppose that $\mathbf{c} \rightsquigarrow_{ss}^{\infty} \mathbf{d}$. Then there is a sequence $\mathbf{b}'_1, \mathbf{b}'_2, \dots$ of states such that $\mathbf{c} \rightarrow^1 \mathbf{b}'_1 \rightarrow^1 \mathbf{b}'_2 \dots$ and, for some subsequence $\mathbf{b}_1, \mathbf{b}_2, \dots$ of $\mathbf{b}'_1, \mathbf{b}'_2, \dots$, $\mathbf{d} = \lim_{i \rightarrow \infty} \mathbf{b}_i$. For each finite $i \geq 1$, for some finite $j \geq i$, $\mathbf{c} \rightsquigarrow^j \mathbf{b}_i$. So, by the finite case shown above, $\mathbf{c} \rightarrow^1 \mathbf{b}_i$. Thus, $\{\mathbf{b}_1, \mathbf{b}_2, \dots\} \subseteq S_{\mathbf{c}}$, where $S_{\mathbf{c}}$ is the set of states straight-line reachable from \mathbf{c} . Since $S_{\mathbf{c}}$ is closed by Lemma 2.11, it contains all its limit points, so $\mathbf{d} \in S_{\mathbf{c}}$ as well, i.e., $\mathbf{c} \rightarrow^1 \mathbf{d}$. \square

Finally, the previous lemmas can be combined to show that at most a constant number of straight line segments (depending on the CRN) are required to reach from any state to any other reachable state. In fact, this holds even for states that are only $\rightsquigarrow_{ss}^{\infty}$ reachable.

THEOREM 2.14. *If $\mathbf{c} \rightsquigarrow_{ss}^{\infty} \mathbf{d}$, then $\mathbf{c} \rightsquigarrow^{m+1} \mathbf{d}$, where $m = \min\{|\Lambda|, |R|\}$. Additionally, there is a constant K , depending only on the CRN, so the path from \mathbf{c} to \mathbf{d} can be chosen so the total flux of all reactions along the path is less than $K\|\mathbf{d} - \mathbf{c}\|$.*

PROOF. First, without loss of generality, we can consider the reduced CRN where we remove all of the reactions that are not used with positive flux in the given path from \mathbf{c} to \mathbf{d} . By Lemma 2.10, we can find a state \mathbf{c}' such that $\mathbf{c} \rightsquigarrow^m \mathbf{c}'$ and $[\mathbf{c}'] = P(\mathbf{c})$. We now show that we can “scale down” the path $\mathbf{c} \rightsquigarrow^m \mathbf{c}'$ such that no reaction occurs more than in the original path $\mathbf{c} \rightsquigarrow \mathbf{d}$, allowing us to complete the path to \mathbf{d} using Lemma 2.13.

First consider the finite case, where $\mathbf{c} \rightsquigarrow^k \mathbf{d}$ for $k < \infty$. We make the following general observation about finite paths: Let $\rho \in \Gamma_l$ be any finite path with segments given by the flux vectors $\mathbf{u}_1 \dots \mathbf{u}_l$, and let $F_{\rho, \alpha}$ be the total flux through the reaction α along ρ , i.e.,

$$F_{\rho, \alpha} = \sum_{i=1}^l \mathbf{u}_i(\alpha).$$

Then, if $\rho = \lambda \rho_1 + (1 - \lambda) \rho_0$, then

$$F_{\rho, \alpha} = \lambda F_{\rho_1, \alpha} + (1 - \lambda) F_{\rho_0, \alpha}.$$

Let $\sigma \in \Gamma_m$ be the path from \mathbf{c} to \mathbf{c}' , let $\sigma_0 \in \Gamma_0 \subseteq \Gamma_m$ be the trivial path staying at \mathbf{c} , and let $\gamma \in \Gamma_k$ be the given path from \mathbf{c} to \mathbf{d} . We can find some small $\varepsilon > 0$ such that

$$\varepsilon F_{\sigma, \alpha} < F_{\gamma, \alpha}$$

for all reactions $\alpha \in R$. As a result, letting $\gamma' = \varepsilon \sigma + (1 - \varepsilon) \sigma_0$, we see that

$$F_{\gamma', \alpha} = \varepsilon F_{\sigma, \alpha} + (1 - \varepsilon) F_{\sigma_0, \alpha} = \varepsilon F_{\sigma, \alpha} < F_{\gamma, \alpha}.$$

Let \mathbf{a} be the state reached via γ' (in particular $\mathbf{c} \rightsquigarrow^m \mathbf{a}$). Since $\mathbf{a} = \varepsilon \mathbf{c}' + (1 - \varepsilon) \mathbf{c}$ and $[\mathbf{c}'] = P(\mathbf{c})$ for the reduced CRN, we also have that $[\mathbf{a}] = P(\mathbf{c})$. Thus, all reactions α of the reduced CRN are applicable at \mathbf{a} , and by Lemma 2.13, the final straight line from \mathbf{a} can be defined by the flux vector $\mathbf{u}_\alpha = F_{\gamma, \alpha} - F_{\gamma', \alpha}$, so $\mathbf{a} + \mathbf{M}\mathbf{u} = \mathbf{d}$. This shows that $\mathbf{a} \rightarrow^1 \mathbf{d}$, so $\mathbf{c} \rightsquigarrow^{m+1} \mathbf{d}$, proving the theorem for the case of finitely many segments.

Now suppose that $\mathbf{c} \rightsquigarrow_{ss}^\infty \mathbf{d}$, and let $\gamma \in \Gamma_\infty$ be the path $\mathbf{c} \xrightarrow{\mathbf{u}_1} \mathbf{c}_1 \xrightarrow{\mathbf{u}_2} \mathbf{c}_2 \dots$ that starts at \mathbf{c} and has an infinite subsequence of states $\mathbf{b}_1, \mathbf{b}_2, \dots$ converging to \mathbf{d} . Because we have assumed without loss of generality that every reaction α occurs with positive flux along γ , we know that

$$\sum_{i=1}^{\infty} \mathbf{u}_i(\alpha)$$

is positive (although it might be infinite). As a result, there is some finite N_α such that

$$\sum_{i=1}^{N_\alpha} \mathbf{u}_i(\alpha) > 0.$$

Let $N = \max_{\alpha \in R} N_\alpha$ be the number of line segments required for each reaction to have had positive flux. The truncation of γ to a path with N segments from \mathbf{c} to \mathbf{c}_N is then a path γ_N where every reaction α occurs with positive flux. By applying the first part of the argument, we can find a state \mathbf{a} with $[\mathbf{a}] = P(\mathbf{c})$ so $\mathbf{c} \rightsquigarrow^m \mathbf{a}$ and $\mathbf{a} \rightarrow^1 \mathbf{c}_N$. But then because $\mathbf{c}_N \rightsquigarrow_{ss}^\infty \mathbf{d}$, we see that $\mathbf{a} \rightsquigarrow_{ss}^\infty \mathbf{d}$, so by Lemma 2.13, we see that $\mathbf{a} \rightarrow^1 \mathbf{d}$. As a result, we conclude that $\mathbf{c} \rightsquigarrow^{m+1} \mathbf{d}$.

To see that we can bound the total flux along the path from \mathbf{c} to \mathbf{d} , first note that by taking ε small enough, we can guarantee both that the total flux along γ' is bounded by $\|\mathbf{d} - \mathbf{c}\|$ and that

$$\|\mathbf{a} - \mathbf{c}\| = \|\mathbf{M}\mathbf{F}_{\gamma'}\| < \|\mathbf{d} - \mathbf{c}\|.$$

By the triangle inequality, this implies that $\|\mathbf{a} - \mathbf{d}\| < 2\|\mathbf{d} - \mathbf{c}\|$. Now by applying Lemma D.4, we see that there's some constant C depending only on the CRN so the flux vector \mathbf{u} of the straight line from \mathbf{a} to \mathbf{d} can be chosen with $\|\mathbf{u}\| < 2C\|\mathbf{d} - \mathbf{c}\|$. Taking $K = 2C + 1$, we see that the flux along the whole path $\mathbf{c} \rightsquigarrow^m \mathbf{a} \rightarrow^1 \mathbf{d}$ is bounded above by $K\|\mathbf{d} - \mathbf{c}\|$. \square

Note that Theorem 2.14 immediately implies that \rightsquigarrow^∞ and $\rightsquigarrow_{ss}^\infty$ are the same relation, since $\mathbf{c} \rightsquigarrow_{ss}^\infty \mathbf{d}$ implies $\mathbf{c} \rightsquigarrow^{m+1} \mathbf{d}$ implies $\mathbf{c} \rightsquigarrow^\infty \mathbf{d}$.

Although the full power of Theorem 2.14 is useful later in Theorem E.1, the most important consequence of Theorem 2.14 is the following result, which we will use repeatedly:

COROLLARY 2.15. *If $\mathbf{c} \rightsquigarrow \mathbf{d}$, then $\mathbf{c} \rightsquigarrow^{m+1} \mathbf{d}$, where $m = \min\{|\Lambda|, |R|\}$.*

PROOF. This follows from Theorem 2.14 and the fact that $\mathbf{c} \rightsquigarrow \mathbf{d}$ implies $\mathbf{c} \rightsquigarrow_{ss}^\infty \mathbf{d}$. \square

Corollary 2.15 will allow us to assume without loss of generality that there are a constant number of line segments between any two states, simplifying many arguments. For example, it is not obvious that the relation \rightsquigarrow^∞ is transitive, since one cannot concatenate two infinite sequences. However, since two finite sequences of segments can be concatenated, the following corollary is immediate:

COROLLARY 2.16. *The relation \rightsquigarrow is transitive.*

The goal of the reachability relation is to capture “what could happen” in chemical reaction networks independently of rates. Thus, it is natural to satisfy several properties: The relation should be reflexive (true for \rightsquigarrow , since $x \rightsquigarrow^0 x$) and transitive (Corollary 2.16). Further, the relation should be *additive* in the intuitive sense that the presence of additional molecules cannot entirely prevent reactions from happening (although in a kinetic model it could effectively slow down reactions due to competition); formally, if $x \rightsquigarrow y$, then $x + c \rightsquigarrow y + c$ for any state c . Additivity is a crucial property of the more standard notion of discrete CRN reachability, used for example in many cases to prove impossibility results for those systems [1, 2, 7, 11, 25]. We also employ additivity of \rightsquigarrow for impossibility results, for example, in the proofs of Lemmas 5.30 and 5.31.¹¹

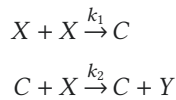
While satisfying these properties is a good start for justifying the reasonableness of \rightsquigarrow , it is natural to wonder whether there are some “reasonable” rate laws that segment-reachability fails to capture, i.e., perhaps some CRN rate law would take x to y even though $x \not\rightsquigarrow y$. In Section 2.4, we define an apparently much more general notion of reachability (Definition 2.22) that captures all commonly studied rate laws, while still respecting the fundamental semantics of reactions. We prove that our reachability relation is in fact identical to it, i.e., x can reach to y under this notion if and only if $x \rightsquigarrow y$ (Theorem 2.27 and Lemma 2.26).

2.4 Generality of Segment Reachability

In this section, we justify that our notion of reachability via straight lines actually corresponds to the most general notion of “being able to get from one state to another,” restricted only by the non-negativity of concentrations, reaction stoichiometry and the need for catalysts—as long as we maintain the causal relationships between the production of species. This notion of reachability admits “time-varying” rate laws where reactions occur according to some arbitrary schedule, which captures situations such as solutions that are not well-mixed, or where physical parameters, such as temperature, change in some arbitrary way. The main result of this section, Theorem 2.27, formalizes this idea and justifies calling segment-reachability simply “reachability” in the rest of this article.

We begin with a review of mass-action kinetics, the most commonly used rate law in chemistry and show the (physically and intuitively obvious but mathematically subtle) features that make it consistent with segment reachability. We then generalize rate-law trajectories to arbitrary “valid rate schedules” and prove that these are exactly captured by segment-reachability.

A CRN with positive rate constants assigned to each reaction defines a mass-action **ODE (ordinary differential equation)** system with a variable for each species, which represents the time-varying concentration of that species. We follow the convention of upper-case species names and lower-case concentration variables. Each reaction contributes one term to the ODEs for each species produced or consumed in it. The term from reaction α appearing in the ODE for x is the product of: the rate constant, the reactant concentrations, and the net stoichiometry of X in α (i.e., the net amount of X produced by α , negative if consumed). For example, the CRN



¹¹Another property of \rightsquigarrow that we use extensively is *scale-invariance*: If $x \rightsquigarrow y$, then $\lambda x \rightsquigarrow \lambda y$ for any $\lambda \geq 0$, which is essentially responsible for the convexity of Lemma 2.7. This does *not* hold for discrete CRN reachability when $\lambda < 1$, even when the scaled discrete states are well-defined, e.g., the reaction $X + X \rightarrow Y$ is applicable in state $x = \{2X\}$ but not in state $0.5x = \{1X\}$ in the discrete model.

corresponds to ODEs:

$$dx/dt = -2k_1x^2 - k_2cx, \quad (2.1)$$

$$dc/dt = k_1x^2, \quad (2.2)$$

$$dy/dt = k_2cx, \quad (2.3)$$

where k_1, k_2 are the rate constants of the two reactions.

Given a CRN $C = (\Lambda, R)$, let $A(\mathbf{d}) : \mathbb{R}_{\geq 0}^\Lambda \rightarrow \mathbb{R}_{\geq 0}^R$ be the rates of all the reactions in state \mathbf{d} as given by the mass-action ODEs. Given an assignment of (strictly) positive rate constants, and an initial state \mathbf{c} , the *mass-action trajectory* is a function $\rho : [0, t_{\max}) \rightarrow \mathbb{R}_{\geq 0}^\Lambda$, where $t_{\max} \in \mathbb{R}_{\geq 0} \cup \{\infty\}$, such that ρ is the solution to $d\rho/dt = \mathbf{M} \cdot A(\rho(t))$ with $\rho(0) = \mathbf{c}$, where t_{\max} is the maximum time, typically ∞ , for which the solution is defined on all of $[0, t_{\max})$. Although beyond the scope of this article, mass-action ODEs are locally Lipschitz, so a CRN admits exactly one mass-action trajectory for a fixed collection of rate constants and initial state \mathbf{c} . Note that for some CRNs (e.g., $2X \rightarrow 3X$), the solution of the ODEs goes to infinite concentration in finite time,¹² and for such CRNs, t_{\max} is finite.

Definition 2.17. Fix an assignment of positive mass-action rate constants. Let \mathbf{c}, \mathbf{d} be two states. We say \mathbf{d} is *mass-action reachable* (with respect to the rate constants) from \mathbf{c} if the associated mass action trajectory ρ obeys $\rho(0) = \mathbf{c}$ and either $\rho(t) = \mathbf{d}$ for some finite $t \geq 0$ or $\lim_{t \rightarrow \infty} \rho(t) = \mathbf{d}$.¹³

To prove Theorem 2.27, we need to introduce the notion of a *siphon* from the Petri net literature. This notion will be used, as well, to prove negative results in Section 5.5.

Definition 2.18. Let $C = (\Lambda, R)$ be a CRN. A *siphon* is a set of species $\Omega \subseteq \Lambda$ such that, for all reactions $\langle \mathbf{r}, \mathbf{p} \rangle \in R$, $[\mathbf{p}] \cap \Omega \neq \emptyset \implies [\mathbf{r}] \cap \Omega \neq \emptyset$, i.e., every reaction that has a product in Ω also has a reactant in Ω .

The following lemma, due to Angeli, De Leenheer, and Sontag [5], shows that this is equivalent to the notion that “the absence of Ω is forward-invariant” under mass-action: If all species in Ω are absent, then they can never again be produced (under mass-action).¹⁴ For the sake of completeness, we give a self-contained proof in Appendix A.

LEMMA 2.19 ([5], PROPOSITION 5.5). *Fix any assignment of positive mass-action rate constants. Let $\Omega \subseteq \Lambda$ be a set of species. Then Ω is a siphon if and only if, for any state \mathbf{c} such that $\Omega \cap [\mathbf{c}] = \emptyset$ and any state \mathbf{d} that is mass-action reachable from \mathbf{c} , $\Omega \cap [\mathbf{d}] = \emptyset$.*

¹²Indeed, the mass-action ODE corresponding to the CRN $2X \rightarrow 3X$ is $\frac{dx}{dt} = x^2$, which is solved by $x(t) = \frac{1}{C-t}$, where $C = 1/x(0)$. This goes to infinity as t approaches C .

¹³Note that a more general definition would say \mathbf{d} is mass-action reachable from \mathbf{c} if there exist positive rate constants such that the trajectory starting at \mathbf{c} passes through or approaches \mathbf{d} . Note, however, that this relation is not transitive: For some CRNs, \mathbf{c} reaches to \mathbf{d} under one set of rate constants, and \mathbf{d} reaches to \mathbf{x} under another set of rate constants, yet no single assignment of rate constants takes the CRN from \mathbf{c} to \mathbf{x} .

¹⁴It is obvious in the discrete CRN model, and in an intuitive physical sense, that if producing a species initially absent causally requires another species also initially absent and vice versa, then neither species can ever be produced. However, it requires care to prove this for mass-action ODEs. Consider the CRN $2X \rightarrow 3X$. The corresponding mass-action ODE is $dx/dt = x^2$, and has the property that starting with $x(0) = 0$, it cannot become positive, i.e., the only solution with $x(0) = 0$ is $x(t) = 0$ for all $t \geq 0$. However, the very similar non-mass-action ODE $dx/dt = x^{1/2}$ has a perfectly valid solution $x(t) = t^2/4$, which starts at 0 but becomes positive, despite the fact that at $t = 0$, $dx/dt = 0$. (Though $x(t) = 0$ for all $t \geq 0$ is another valid solution.) The difference is that mass-action polynomial rates are locally Lipschitz (have bounded rates of change, unlike $x^{1/2}$, whose derivative goes to ∞ as $x \rightarrow 0$) and so are guaranteed to have a unique solution by the Picard-Lindelöf theorem.

We show that the same holds true for segment-reachability. Due to the discrete nature of segment-reachability, the proof is more straightforward than that of Lemma 2.19. It follows the same essential structure one would use to prove this in the discrete CRN model: If the siphon Ω is absent, then no reaction with a reactant in Ω can be the next reaction to fire, so by the siphon property, no species in Ω is produced in the next step.

LEMMA 2.20. *Let $\Omega \subseteq \Lambda$ be a set of species. Then Ω is a siphon if and only if, for any state \mathbf{c} such that $\Omega \cap [\mathbf{c}] = \emptyset$ and any state \mathbf{d} such that $\mathbf{c} \rightsquigarrow \mathbf{d}$, $\Omega \cap [\mathbf{d}] = \emptyset$.*

PROOF. To see the forward direction, suppose Ω is a siphon, let \mathbf{c} be a state such that $[\mathbf{c}] \cap \Omega = \emptyset$, and let \mathbf{d} be such that $\mathbf{c} \rightsquigarrow \mathbf{d}$. By Theorem 2.15, there is a finite path γ such that the n th line segment is between states $\mathbf{x}_{n-1}(\gamma)$ and $\mathbf{x}_n(\gamma)$, with $\mathbf{c} = \mathbf{x}_0(\gamma)$ and $\mathbf{d} = \mathbf{x}_{m+1}(\gamma)$. Assume inductively that $[\mathbf{x}_{n-1}(\gamma)] \cap \Omega = \emptyset$; then no reaction applicable at $\mathbf{x}_{n-1}(\gamma)$ has reactants in Ω . So, by definition of siphon, no reaction applicable at $\mathbf{x}_{n-1}(\gamma)$ has products in Ω , and $[\mathbf{x}_n(\gamma)] \cap \Omega = \emptyset$ as well. Therefore, $\mathbf{d} \cap \Omega = \emptyset$. This shows the forward direction.

To show the reverse direction, suppose that Ω is not a siphon. Then there is a reaction $\alpha = \langle \mathbf{r}, \mathbf{p} \rangle$ such that $[\mathbf{p}] \cap \Omega \neq \emptyset$, but $[\mathbf{r}] \cap \Omega = \emptyset$. Then from any state \mathbf{c} such that $[\mathbf{c}] = \Lambda \setminus \Omega$ (i.e., all species not in Ω are present), all reactants of α are present, so α is applicable. Running α produces S , hence results in a state \mathbf{d} such that $\mathbf{c} \rightsquigarrow \mathbf{d}$ with $\Omega \cap [\mathbf{d}] \neq \emptyset$, since $S \in \Omega$. \square

Recall that $P(\mathbf{c})$ represents the set of species producible from state \mathbf{c} . The next lemma shows that the set of species that cannot ever be produced from a given state is a siphon.

LEMMA 2.21. *If \mathbf{c} is any state, then $\Omega = \Lambda \setminus P(\mathbf{c})$ is a siphon.*

PROOF. By Lemma 2.10 there is a state \mathbf{c}' that is segment reachable from \mathbf{c} with all of the species in $P(\mathbf{c})$ present. If Ω were not a siphon, then there would be a reaction α that produced a species S of Ω such that all of the reactants of α would be contained in $\Lambda \setminus \Omega = P(\mathbf{c})$. This implies that α would be applicable at \mathbf{c}' , so S would be in $P(\mathbf{c})$, giving a contradiction. \square

The main result of this section is Theorem 2.27, which justifies that our (seemingly limited) notion of reachability via straight lines is actually quite general. To state the theorem, we define a very general notion of “reasonable rate laws,” which are essentially schedules of rates to assign to reactions over time. All known rate laws, such as mass-action, Michaelis-Menten, Hill function kinetics, as well as our own nondeterministic notion of adversarial rates following straight lines (segment reachability, Definition 2.3), obey this definition. (We justify this below explicitly for mass-action and Definition 2.3, but it is straightforward to verify in the other cases.)

Recall that R is the set of all reactions in some CRN, and Λ is the set of its species.

Definition 2.22. A *rate schedule* is a function $\mathbf{f} : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}^R$. We interpret $\mathbf{f}_\alpha(t)$ to be the rate, or instantaneous flux, at which reaction α occurs at time t . Given a state $\mathbf{c} \in \mathbb{R}_{\geq 0}^\Lambda$, we say \mathbf{f} is a *valid rate schedule starting at \mathbf{c}* if:

- (1) **(Total reaction fluxes are well-defined).** For each $\alpha \in R$, \mathbf{f}_α is (locally Lebesgue) integrable: for each time $t \geq 0$, $\mathbf{F}_\alpha(t) = \int_0^t \mathbf{f}_\alpha(t) dt$ is well-defined and finite (although $\int_0^\infty \mathbf{f}_\alpha(t) dt$ may be infinite). Let $\mathbf{F}(t)$ be the vector in $\mathbb{R}_{\geq 0}^R$ whose α coordinate is $\mathbf{F}_\alpha(t)$, which represents the total amount of each reaction flux that has happened by time t .¹⁵

¹⁵An alternative to Definition 2.22 would start with a differentiable trajectory ρ and total flux \mathbf{F} (related via $\rho(t) = \mathbf{M} \cdot \mathbf{F}(t) + \mathbf{c}$) and define $\mathbf{f} = d\mathbf{F}/dt$. However, requiring differentiable ρ and \mathbf{F} rules out many natural cases, such as the rate schedules implicit in segment-reachability (Definition 2.2), whose trajectories are not differentiable at cusp points \mathbf{b}_i in between straight lines and whose rate schedules are not even continuous.

Define the *trajectory* $\rho : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}^\Lambda$ of \mathbf{f} starting at \mathbf{c} for all $t \geq 0$ by $\rho(t) = \mathbf{M} \cdot \mathbf{F}(t) + \mathbf{c}$, which represents the state of the CRN at time t .

- (2) **(Positive-rate reactions require their reactants present).** For all times $t \geq 0$ and reactions $\alpha \in R$, if $\mathbf{f}_\alpha(t) > 0$, then α is applicable in $\rho(t)$.
- (3) **(Absence of siphons is forward-invariant).** For every siphon $\Omega \subseteq \Lambda$, if $\Omega \cap [\rho(t)] = \emptyset$ for some time t , then $\Omega \cap [\rho(t')] = \emptyset$ for all times $t' \geq t$.

Definition 2.23. We say that a state \mathbf{d} is *reachable from a state \mathbf{c} by a valid rate schedule* if there is a valid rate schedule \mathbf{f} starting at \mathbf{c} , with trajectory ρ , such that either $\mathbf{d} = \rho(t_f)$ for some $t_f < \infty$, or $\mathbf{d} = \lim_{t \rightarrow \infty} \rho(t)$. In the first case, we say that \mathbf{d} is reached in finite time.

We note that because \mathbf{f} is Lebesgue integrable, by Reference [43, Theorem 6.11], $\mathbf{F}_\alpha(t)$ is locally absolutely continuous.

Although Definition 2.22 explicitly constrains the states $\rho(t)$ to be non-negative, the non-negativity of $\rho(t)$ actually follows from condition (2).¹⁶

Conditions (2) and (3) may appear redundant, but in fact each can be obeyed while the other is violated.

For example, consider the reaction $\alpha : X \rightarrow 2X$, starting in the state $\{0X\}$, with invalid rate schedule $\mathbf{f}_\alpha(t) = t$, with trajectory $\rho_X(t) = t^2/2$. Since the rate $\mathbf{f}_\alpha(0)$ is 0, this vacuously satisfies (2) at time 0, and, since $\rho_X(t) > 0$ for $t > 0$ (X is present at all positive times), (2) is also satisfied for positive times. However, (3) is violated, since $\{X\}$ is a siphon absent at time 0 but present at future times. This example also demonstrates why condition (3) is required to satisfy our intuitive understanding of reasonable rate laws respecting “causality of production” among species: With only the reaction $X \rightarrow 2X$, the only way to produce more X is already to have some X .

To see the other case, take reactions $\alpha : X \rightarrow C$ and $\beta : C + X \rightarrow C + Y$, starting in state $\{1X, 1Y, 0C\}$. Consider the invalid rate schedule $\mathbf{f}_\alpha(t) = 0$ for all t , $\mathbf{f}_\beta(t) = 1$ for $0 \leq t \leq 1/2$, and $\mathbf{f}_\beta(t) = 0$ for $t > 1/2$, i.e., run only β , until X is half gone. This violates (2), since β occurs without its reactant C present. However, the only set of species absent along this trajectory is $\{C\}$, which is not a siphon, since reaction α has C as a product but not a reactant, so (3) is vacuously satisfied.

The next lemma shows that the most commonly used rate law, mass-action, gives a valid rate schedule and trajectory according to Definition 2.22. Recall that $\mathbf{A}(\rho(t)) : \mathbb{R}_{\geq 0}^\Lambda \rightarrow \mathbb{R}_{\geq 0}^R$ represents the rates of all the reactions in state $\rho(t)$ as given by the mass-action ODEs. For instance, for our mass-action example at the beginning of this section, the function $\mathbf{f}(t) = \mathbf{A}(\rho(t))$ corresponds to the ODEs of Equations ((2.1))–((2.3)) when written as $d\rho/dt = \mathbf{M} \cdot \mathbf{f}(t)$.

LEMMA 2.24. *Suppose we fix an assignment of positive mass-action rate constants for a given CRN as well as an initial state \mathbf{c} . Suppose that the associated mass action trajectory ρ is defined for all time. Then $\mathbf{f} : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}^R$ such that $\mathbf{f}(t) = \mathbf{A}(\rho(t))$ is a valid rate schedule whose trajectory is ρ .*

¹⁶Consider a rate schedule that takes concentrations negative: for instance, starting with $1X$ and applying reaction $\alpha : X \rightarrow Y$ with $\mathbf{F}_\alpha(t) = 2$ for some $t > 0$. To see that this contradicts condition (2) when we naturally generalize notation $[\mathbf{c}]$ to possibly negative \mathbf{c} (for any $\mathbf{c} \in \mathbb{R}^\Lambda$, $[\mathbf{c}] = \{S \in \Lambda \mid \mathbf{c}(S) > 0\}$), suppose that $\rho_X(t') < 0$ for some species X at some time t' . Let t_0 be the supremum of all the times less than t' where $\rho_X(t) \geq 0$. Recall $\mathbf{F}_\alpha(t)$ is a locally absolutely continuous (and therefore continuous) function. Thus, $\rho(t)$ is also a continuous function so $\rho_X(t_0) \geq 0$. Moreover, for all $t_0 < t < t'$, we know that $\rho_X(t) < 0$ by our choice of t_0 . So, by condition (2) $\mathbf{f}_\alpha(t) = 0$ for all α where X is a reactant, and therefore (recall $\mathbf{M}(X, \alpha)$ means the net consumption of X in reaction α)

$$\rho_X(t') = \rho_X(t_0) + \sum_{\alpha \in R} \mathbf{M}(X, \alpha) \int_{t_0}^{t'} \mathbf{f}_\alpha(t) dt \geq \rho_X(t_0) \geq 0,$$

a contradiction, since $\rho(t')$ was assumed to be negative. See also Reference [26, Proposition 2.8], where the term *strict* is equivalent to condition (2).

PROOF. First, note that because $\rho(t)$ is a real analytic function, $A(\rho(t))$ is necessarily also real analytic, and therefore locally integrable, so condition (1) of Definition 2.22 is satisfied. Let $\tilde{\rho}(t)$ be the trajectory associated with the rate schedule f . Then, because $\rho(t)$ is a solution to the mass-action ODEs with initial state c ,

$$\begin{aligned}\tilde{\rho}(t) &= c + M \int_0^t f(t) dt \\ &= c + \int_0^t M \cdot A(\rho(t)) dt \\ &= c + \int_0^t \frac{d}{dt} \rho(t) dt \\ &= c + (\rho(t) - c) \\ &= \rho(t).\end{aligned}$$

Because $A(\rho(t))$ can only be positive when $\rho_X(t) > 0$ for all reactants X of α , we see that $f_\alpha(t) > 0$ implies that $[\tilde{\rho}(t)] = [\rho(t)]$ contains all of the reactants of α . Therefore, condition (2) of Definition 2.22 is satisfied. Finally, condition (3) of Definition 2.22 is satisfied by Lemma 2.19. \square

We say that a rate schedule $f : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}^R$ is *finite* if there is $t_0 \geq 0$ such that $f(t) = \mathbf{0}$ for all $t \geq t_0$, i.e., reactions eventually stop occurring. The following observation is straightforward to verify, showing that the concatenation of two valid rate schedules, with the first finite, is also a valid rate schedule.

OBSERVATION 2.25. *If f, g are valid rate schedules, with f finite such that $f(t) = \mathbf{0}$ for all $t \geq t_0$, then h defined by $h(t) = f(t)$ for $0 \leq t \leq t_0$ and $h(t) = g(t - t_0)$ for $t > t_0$, is a valid rate schedule.*

The next lemma shows essentially that our definition of segment-reachability creates a valid rate schedule.

LEMMA 2.26. *If $c \rightsquigarrow d$, then d is reachable from c by a valid rate schedule.*

PROOF. Since $c \rightsquigarrow d$, by Theorem 2.15, we know that $c \rightsquigarrow^{m+1} d$. Using induction and Observation 2.25, it suffices to verify that the rates defined by the straight-line reachability relation $c \rightarrow_u^1 d$ describe a valid rate schedule, since the rate schedules given by \rightsquigarrow^{m+1} are simply concatenations of these. Define $f : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}^R$ by $f(t) = u(\alpha)$ for $0 \leq t < 1$ and $f(t) = \mathbf{0}$ for all $t \geq 1$. (In other words, for one unit of time, run the reactions at constant rates described by u .) Then f is piecewise constant, and therefore integrable, so condition (1) of Definition 2.22 is satisfied. Next note that for all $t \geq 1$, since $f_\alpha(t) = 0$ for all α , condition (2) of Definition 2.22 holds vacuously, and condition (3) holds because $\rho(t) = \rho(t')$ for any $t' > t$. Now let $t < 1$. Observe that

$$\begin{aligned}\rho(t) &= c + M \int_0^t f(T) dT \\ &= c + M \int_0^t u(T) dT \\ &= c + tMu \\ &= (1-t)c + t(c + Mu) \\ &= (1-t)c + td.\end{aligned}$$

Since d is a state (and thus non-negative on all species) and $t < 1$, every species present with positive concentration in c is present with positive concentration in $\rho(t)$. Thus, all reactions applicable at c are also applicable at $\rho(t)$, so condition (2) of Definition 2.22 is also satisfied. Finally by

Lemma 2.20, we see that condition (3) of Definition 2.22 is satisfied and therefore \mathbf{f} is a valid rate schedule starting at \mathbf{c} . Since $\boldsymbol{\rho}(1) = \mathbf{d}$, we see that \mathbf{d} is reachable from \mathbf{c} by a valid rate schedule. \square

Finally, we have the main result of this section, which shows that segment reachability is as general as any valid rate schedule.

THEOREM 2.27. *Given two states \mathbf{c} and \mathbf{d} , \mathbf{d} is reachable from \mathbf{c} by a valid rate schedule if and only if $\mathbf{c} \rightsquigarrow \mathbf{d}$.*

PROOF. Lemma 2.26 establishes the reverse direction. To see the forward direction, let \mathbf{f} be the valid rate schedule from \mathbf{c} to \mathbf{d} , and define \mathbf{F} and $\boldsymbol{\rho}$ for \mathbf{f} as in Definition 2.22.

First, suppose \mathbf{d} is reached in finite time $t_f \in \mathbb{R}_{\geq 0}$, i.e., $\mathbf{d} = \boldsymbol{\rho}(t_f)$.

We say that a reaction α occurs with positive flux if $\mathbf{F}_\alpha(t_f) > 0$. Let $R_\rho = \{\alpha \in R \mid \mathbf{F}_\alpha(t_f) > 0\}$ be the reactions that occur with positive flux along the trajectory $\boldsymbol{\rho}$, and let $\Lambda_\rho = \{S \in \Lambda \mid \rho_S(t) > 0 \text{ for some } 0 \leq t \leq t_f\}$ be the species that are present with positive concentration at some point along the trajectory $\boldsymbol{\rho}$.

Consider removing species not in Λ_ρ and reactions not in R_ρ . We claim that the pair (Λ_ρ, R_ρ) is a well-defined CRN, as defined in Section 2.1, because every reactant and product in R_ρ is in Λ_ρ . To see why, let $S \in \Lambda \setminus \Lambda_\rho$ (i.e., $\rho_S(t) = 0$ for all $0 \leq t \leq t_f$), let R_S be the reactions with S as a reactant, and let P_S be the reactions with S as a product; we must show $R_S \cap R_\rho = \emptyset$ and $P_S \cap R_\rho = \emptyset$. By Definition 2.22 part (2), no reaction in R_S has positive flux, so $R_S \cap R_\rho = \emptyset$. Since no reaction in R_S has positive flux, no reaction in P_S can have positive flux or else S would be produced with no reaction to consume it, contradicting $\rho_S(t_f) = 0$, so $P_S \cap R_\rho = \emptyset$.

Now, we claim that every species in this reduced CRN (Λ_ρ, R_ρ) is segment-producible from \mathbf{c} , i.e., $\mathbf{P}(\mathbf{c}) = \Lambda_\rho$. If not, then $\Omega = \Lambda_\rho \setminus \mathbf{P}(\mathbf{c})$ is non-empty. Letting S^* be some element of Ω , we know that S^* has positive concentration along $\boldsymbol{\rho}$ by our construction of the reduced CRN. However, by Lemma 2.21, Ω is a siphon. Since \mathbf{c} is zero on Ω and $S^* \in \Omega$, this violates Definition 2.22 part (3).

Since every species in our reduced CRN is segment-producible from \mathbf{c} , by Lemma 2.10, we can construct a state \mathbf{c}' segment-reachable from \mathbf{c} where all of the species in the reduced CRN are present simultaneously. Since every reaction in R_ρ is applicable at \mathbf{c}' , the remainder of the proof is similar to the proof for the finite case of Theorem 2.15: by “scaling down” the path from \mathbf{c} to \mathbf{c}' , there is a state \mathbf{a} such that $\mathbf{c} \rightsquigarrow^m \mathbf{a}$ (where $m = \min\{|\Lambda_\rho|, |R_\rho|\}$), $[\mathbf{a}] = \mathbf{P}(\mathbf{c}) = \Lambda_\rho$, and $\mathbf{a} \rightarrow^1 \mathbf{d}$. Thus, $\mathbf{c} \rightsquigarrow^{m+1} \mathbf{d}$. This handles the case that \mathbf{d} is reached in finite time.

However, suppose that \mathbf{d} is not reached from \mathbf{c} in finite time, but instead $\mathbf{d} = \lim_{t \rightarrow \infty} \boldsymbol{\rho}(t)$. This case is similar to the proof of the infinite case of Theorem 2.15. By definition of R_ρ , for each reaction $\alpha \in R_\rho$, $\mathbf{F}_\alpha(t) > 0$ for some $t > 0$. As a result, there is some finite t_α such that $\mathbf{F}_\alpha(t_\alpha) > 0$. Let $t_{\text{pos}} = \max\{t_\alpha \mid \alpha \in R_\rho\}$, noting that $\mathbf{F}(t_{\text{pos}}) > 0$ for all $\alpha \in R_\rho$, i.e., each reaction has occurred by time t_{pos} . Let $\mathbf{b} = \boldsymbol{\rho}(t_{\text{pos}})$.

By applying the first part of the argument, we can find a state \mathbf{a} with $[\mathbf{a}] = \mathbf{P}(\mathbf{c})$ such that $\mathbf{c} \rightsquigarrow^m \mathbf{a}$ and $\mathbf{a} \rightarrow^1 \mathbf{b}$. Now let $\mathbf{d}_t = \boldsymbol{\rho}(t)$ for every time $t > t_{\text{pos}}$. Because $\boldsymbol{\rho}$ restricted to $[t_{\text{pos}}, t]$ gives a finite trajectory from \mathbf{b} to \mathbf{d}_t , we know by the first part of the argument that $\mathbf{b} \rightsquigarrow \mathbf{d}_t$, so $\mathbf{a} \rightsquigarrow \mathbf{d}_t$. By Lemma 2.13, we see that $\mathbf{a} \rightarrow^1 \mathbf{d}_t$. Since $\mathbf{d} = \lim_{t \rightarrow \infty} \mathbf{d}_t$, we see by Lemma 2.11 that $\mathbf{a} \rightarrow^1 \mathbf{d}$. As a result, we conclude that $\mathbf{c} \rightsquigarrow \mathbf{d}$. \square

Recall mass-action trajectories correspond to valid rate schedules by Lemma 2.24. Thus, Theorem 2.27 implies the following corollary, which intuitively says that if a state is reachable via a mass-action trajectory (even in the limit of infinite time), then it is segment-reachable.¹⁷

¹⁷Although Lemma 2.24 has the precondition that the mass-action trajectory be defined for all time, states reached in finite time by diverging mass-action CRNs can also be segment-reached. For example, for the CRN $2X \rightarrow 3X$ (with rate

COROLLARY 2.28. *Fix an assignment of positive mass-action rate constants for a given CRN. Let \mathbf{c}, \mathbf{d} be two states such that \mathbf{d} is mass-action reachable from \mathbf{c} . Then $\mathbf{c} \rightsquigarrow \mathbf{d}$.*

3 STABLE COMPUTATION

We now use segment-reachability (Definition 2.3) to formalize what it means for a CRN to *stably compute* a function (Definition 3.2). The notion of stable computation is motivated by, and is essentially identical to, the definition of stable computation for population protocols and discrete CRNs [7, 18].

In this section, we justify stable computation by arguing for *necessity*: CRCs that we can reasonably call “rate-independent” must obey stable computation. Thus, stable computation is immediately useful for negative (impossibility) results: Showing a function cannot be stably computed implies it is not rate-independent in the desired intuitive sense. In the next section (Section 4), we address the other (sufficiency) direction and connect stable computation to another notion of computation based on convergence in the limit as $t \rightarrow \infty$ that provides very strong guarantees for the desired rate-independent behavior of our constructions. Based on this connection, stable computation is taken as the primary definition of rate-independent computation in this work.

First, to formally define what it means for such a CRN to compute a function in any sense, we single out some aspects of the CRN as semantically meaningful. Formally, a **chemical reaction computer (CRC)** is a tuple $C = (\Lambda, R, \Sigma, \Gamma)$, where (Λ, R) is a CRN, $\Sigma \subseteq \Lambda$, written as $\Sigma = \{X_1, \dots, X_k\}$,¹⁸ is the set of *input species*, and $\Gamma \subseteq \Lambda \setminus \Sigma$ is the set of *output species*. Input and output values can also be encoded indirectly via combinations of species. An important encoding for the purposes of this article will be the dual-rail representation (discussed in Section 5), which can handle both positive and negative quantities and allows for easier composition of CRN “modules.” Since we focus on single-output functions, we will have either a single output species $\Gamma = \{Y\}$, or in the case of dual-rail computation, two output species $\Gamma = \{Y^+, Y^-\}$.

We now define output-stable states and stable computation. Intuitively, output-stable states are “ideal” output states for rate-independent computation: The output is correct and no rate law can change it. Stable computation is then defined with respect to output-stable states by requiring that the correct output-stable state remains reachable no matter what “devious rate laws” may do. Although it is not obvious that the notion of output-stable states remains pertinent when transferred from the discrete setting to the continuous one (see the discussion at the beginning of Section 4), Theorem 3.3 below and the subsequent results of Section 4 show that output stability remains crucial.

Definition 3.1. A state $\mathbf{o} \in \mathbb{R}_{\geq 0}^\Lambda$ is *output-stable* if, for all \mathbf{o}' such that $\mathbf{o} \rightsquigarrow \mathbf{o}'$, $\mathbf{o} \upharpoonright \Gamma = \mathbf{o}' \upharpoonright \Gamma$, i.e., once \mathbf{o} is reached, no reactions can change the concentration of any output species.

Note that for a single output species Y , Definition 3.1 says that $\mathbf{o}(Y) = \mathbf{o}'(Y)$ for all \mathbf{o}' such that $\mathbf{o} \rightsquigarrow \mathbf{o}'$. For the sake of brevity and readability, subsequently, we will state many definitions and formal theorem/lemma statements assuming there is only a single output species Y . In each case, there is a straightforward modification of the definition or result, so it applies to CRCs with multiple output species as well.

constant 1) starting in $\{1X\}$, which diverges as $t \rightarrow 1$, all states on the trajectory prior to time $t = 1$ are segment reachable: For each such state, we can construct a valid rate schedule that obeys mass-action until reaching that state and then is constant for all later time.

¹⁸We assume a canonical ordering of $\Sigma = \{X_1, \dots, X_k\}$ so a vector $\mathbf{x} \in \mathbb{R}_{\geq 0}^k$ (i.e., an input to f) can be viewed equivalently as a state $\mathbf{x} \in \mathbb{R}_{\geq 0}^\Sigma$ of C (i.e., an input to C). Note that we have defined valid initial states to contain only the input species Σ ; other species must have initial concentration 0. Our results would change slightly if we relaxed this assumption—see Section 6.2.

Definition 3.2. Let $f : \mathbb{R}_{\geq 0}^k \rightarrow \mathbb{R}_{\geq 0}$ be a function and let C be a CRC. We say that C *stably computes* f if, for all $\mathbf{x} \in \mathbb{R}_{\geq 0}^k$, for all \mathbf{c} such that $\mathbf{x} \rightsquigarrow \mathbf{c}$, there exists an output-stable state \mathbf{o} such that $\mathbf{c} \rightsquigarrow \mathbf{o}$ and $\mathbf{o}(Y) = f(\mathbf{x})$.

To extend our results to functions with l outputs, we can compute l separate functions $f_j : \mathbb{R}_{\geq 0}^k \rightarrow \mathbb{R}_{\geq 0}$ for $1 \leq j \leq l$ with l independent CRCs and then combine them into a single CRC with l output species. In particular, we can use reactions like $X_i \rightarrow X_i^1 + \dots + X_i^l$ to copy input X_i to each of the l CRCs.

We now capture in a theorem the intuition that for a CRC to compute a function rate-independently in any reasonable sense, it must stably compute the function. The theorem says that if a CRC does not stably compute, then, no matter what you do, an adversary can “fight back” and make the output substantially (ϵ) wrong. The proof uses the definitions of partial states and partial reachability, as well as Theorem E.1, which are in Appendix E.

THEOREM 3.3. *Suppose a CRC does not stably compute $f : \mathbb{R}_{\geq 0}^k \rightarrow \mathbb{R}_{\geq 0}$. Then there is $\epsilon > 0$, input state \mathbf{x} and state \mathbf{z} reachable from \mathbf{x} such that for all \mathbf{o} reachable from \mathbf{z} there is \mathbf{o}' reachable from \mathbf{o} such that $|\mathbf{o}'(Y) - f(\mathbf{x})| > \epsilon$.*

PROOF. We prove the contrapositive. Suppose that for all $\epsilon > 0$, for any given input state \mathbf{x} and state \mathbf{z} such that $\mathbf{x} \rightsquigarrow \mathbf{z}$, there exists a state \mathbf{o} such that $\mathbf{z} \rightsquigarrow \mathbf{o}$ and for all \mathbf{o}' such that $\mathbf{o} \rightsquigarrow \mathbf{o}'$, $|\mathbf{o}'(Y) - f(\mathbf{x})| \leq \epsilon$.

For any input state \mathbf{x} and any \mathbf{z} reachable from \mathbf{x} , first, we argue that there is an infinite sequence of states $\mathbf{x}_{1/2}, \mathbf{x}_{1/3}, \mathbf{x}_{1/4}, \dots$ such that $\mathbf{x} \rightsquigarrow \mathbf{z} \rightsquigarrow \mathbf{x}_{1/2} \rightsquigarrow \mathbf{x}_{1/3} \rightsquigarrow \mathbf{x}_{1/4} \rightsquigarrow \dots$, and for all $n \geq 2$, for all \mathbf{o}' such that $\mathbf{x}_{1/n} \rightsquigarrow \mathbf{o}'$, $|\mathbf{o}'(Y) - f(\mathbf{x})| \leq 1/n$. In other words, there is a sequence of states we can visit, where the adversary has less and less freedom to push the output away from the target value $f(\mathbf{x})$. This is true by induction on n , choosing $\epsilon = 1/(n+1)$, $\mathbf{z} = \mathbf{x}_{1/n}$ and $\mathbf{o} = \mathbf{x}_{1/(n+1)}$ in the above assumption.

By the definition of $\mathbf{x}_{1/n}$, we see that $|\mathbf{x}_{1/n}(Y) - f(\mathbf{x})| \leq 1/n$, so $\mathbf{x}_{1/n}(Y)$ converges to $f(\mathbf{x})$ as $n \rightarrow \infty$. Let $\Delta = \{Y\}$ and let $\mathbf{p} \in \mathbb{R}_{\geq 0}^\Delta$ be the partial state with $\mathbf{p}(Y) = f(\mathbf{x})$. Then, we see that $\mathbf{z} \rightsquigarrow_{ss}^\infty \mathbf{p}$ via $(\mathbf{x}_{1/i})_{i=2}^\infty$, so by Theorem E.1, we can find a partition of Δ into Δ_b and Δ_u with $Y \in \Delta_b$, a state \mathbf{y} (d in Theorem E.1), and a subsequence $(\mathbf{x}'_i)_i$ of $(\mathbf{x}_{1/n})_n$ so $\mathbf{z} \rightsquigarrow \mathbf{y}$ and the subsequence (\mathbf{x}'_i) has the property that $\mathbf{x}'_i(S) \rightarrow \infty$ for all $S \in \Delta_u$ and $\mathbf{x}'_i(S) \rightarrow \mathbf{y}(S)$ for all $S \in \Delta_b$. Note that because $Y \in \Delta_b$, we have $\mathbf{y}(Y) = \lim_{i \rightarrow \infty} \mathbf{x}'_i(Y) = f(\mathbf{x})$.

We now claim that \mathbf{y} is also output-stable, which is sufficient to prove the lemma as follows: Since \mathbf{y} is correct ($\mathbf{y}(Y) = f(\mathbf{x})$) and reachable from \mathbf{z} , an arbitrary state reachable from input state \mathbf{x} , this establishes that the CRC in fact stably computes f .

Suppose for the sake of contradiction that \mathbf{y} is not output-stable. Then there is some \mathbf{y}' with $\mathbf{y} \rightsquigarrow \mathbf{y}'$ and $|\mathbf{y}'(Y) - \mathbf{y}(Y)| > \epsilon$. Let $\mathbf{z}_i = \mathbf{x}'_i - \mathbf{y}/2 + \mathbf{y}'/2$. Because $\mathbf{x}'_i(S) \rightarrow \mathbf{y}(S)$ for all $S \in \Delta_b$ and $\mathbf{x}'_i(S) \rightarrow \infty$ for all $S \in \Delta_u$, there is some N so for all $i > N$, we have $\mathbf{x}'_i \geq \mathbf{y}/2$. Then, by additivity of \rightsquigarrow , for all $i > N$,

$$\mathbf{z}'_i = (\mathbf{x}'_i - \mathbf{y}/2) + \mathbf{y}/2 \rightsquigarrow (\mathbf{x}'_i - \mathbf{y}/2) + \mathbf{y}'/2 = \mathbf{z}_i,$$

where $|\mathbf{z}_i(Y) - \mathbf{x}'_i(Y)| = |\mathbf{y}'(Y) - \mathbf{y}(Y)|/2 > \epsilon/2$. Since (\mathbf{x}'_i) is a subsequence of $(\mathbf{x}_{1/n})$, if we take i large enough so \mathbf{x}'_i is \mathbf{x}_{1/n_i} with $1/n_i < \epsilon/4$, then we see that

$$|\mathbf{z}_i(Y) - f(\mathbf{x})| \geq |\mathbf{z}_i(Y) - \mathbf{x}'_i(Y)| - |\mathbf{x}'_i(Y) - f(\mathbf{x})| \geq \epsilon/2 - 1/n_i > 1/n_i,$$

but \mathbf{z}_i is reachable from \mathbf{x}_{1/n_i} , which by definition can only reach states \mathbf{z} with $|\mathbf{z}(Y) - f(\mathbf{x})| \leq 1/n_i$, giving a contradiction. \square

In some places, we will talk about CRCs with multiple output species representing multi-valued functions:

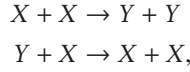
COROLLARY 3.4. *Suppose a CRC does not stably compute $f : \mathbb{R}_{\geq 0}^k \rightarrow \mathbb{R}_{\geq 0}^l$. Then there is an $\epsilon > 0$, an input state \mathbf{x} and a state \mathbf{z} reachable from \mathbf{x} so for all \mathbf{o} reachable from \mathbf{z} there is \mathbf{o}' reachable from \mathbf{o} such that $|\mathbf{o}'(Y_i) - f(\mathbf{x})_i| > \epsilon$ for some $1 \leq i \leq l$.*

PROOF. This is similar to the proof of Theorem 3.3, but make $\Delta = \{Y_1, \dots, Y_l\}$ the set of all output species. \square

4 FAIR COMPUTATION

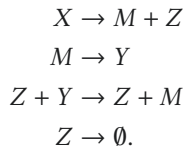
In the discrete model of CRN kinetics, if the set of states reachable from any input state is finite (i.e., the molecular counts are bounded as a function of the input state), then stable computation as in Definition 3.2 (the correct output-stable state is always *reachable*) is equivalent to the condition that the CRC is correct under standard stochastic kinetics with probability 1 (the correct output state state is actually *reached*) [7]. In the continuous CRN model, however, it might seem that the idea of stable computation is not strong enough to achieve intuitively “rate-independent” computation. There are at least two reasons for the concern.

First, it is possible that the output-stable state is always reachable but the mass-action trajectory does not converge to it. For example, consider the following CRC stably computing the identity function $f(x) = x$:



with X as the input species and Y as the output species. From any reachable state, we can reach the output-stable state with all X converted to Y . However, the mass-action trajectory converges to a dynamic equilibrium with $k_2/(2k_1 + k_2)$ fraction of X , where k_1, k_2 are rate constants of the two reactions.¹⁹ This shows that in general, showing that a CRC stably computes is not sufficient to claim that it computes rate-independently in any intuitive sense.

The second difficulty lies with the notion of output-stable states. While the notion of output-stable states is natural for discrete CRNs where we want the system to actually reach that state to end the computation, convergence to the output-stable state in continuous CRNs will typically be only in the limit $t \rightarrow \infty$. For example, consider the following CRC:



The CRC stably computes $f(x) = x$, since from any reachable state, we can reach the output-stable state \mathbf{o} with $f(x)$ amount of Y by converting any remaining X to M , converting any remaining M to Y and completely draining Z . Note that \mathbf{o} is output-stable, since without Z , Y cannot be converted back to M . Further, under mass-action kinetics (for any choice of rate constants), the CRC converges to \mathbf{o} , since as Z drains, the rate of the third reaction converges to 0. However, at every finite time $0 < t < \infty$ in the mass-action trajectory, since Z is present, the state with zero

¹⁹The discrepancy between stable computation and correctness under mass-action kinetics shows a major difference between the discrete and continuous CRN models. In the example above, with n total molecules, the discrete CRN model does a random walk on the number of X that is biased toward the dynamic equilibrium point $nk_2/(2k_1 + k_2)$. Despite the bias upward when X is below this value, there is always a positive probability to decrease X , so with probability 1, X will eventually reach 0.

amount of Y is reachable, so an adversary could substantially perturb the output. Thus, one would not call this CRC rate-independent to adversarial perturbations.

While in the previous section, we argued that stable computation is *necessary* for an intuitive notion of rate-independent computation, the examples above seem to suggest that it is not *sufficient* and that basing rate-independent computation entirely on stable computation could be ill-founded.

In this section, we develop an alternative approach to defining a very strong notion of “rate-independent” computation not based on stable computation, an approach we term “fair computation.” The approach is based on delineating a very broad class of rate laws, possibly adversarial, that still lead to convergence to the correct output. Based on the previous section (Theorem 3.3), it is not surprising that CRCs that fail to stably compute also fail to fairly compute. What is more surprising, however, is that there is a strong connection in the other direction for a class of CRCs (feedforward)—for these CRCs stable computation *implies* fair computation. All our constructions will be in this class; thus, we obtain very strong rate-independence guarantees in the positive results part of this work. Combined with the results of the previous section, stable computation can thus be used as an easy-to-analyze proxy for proving both positive and negative results on rate-independent computation.

Intuitively, a CRC is said to fairly compute if it converges to the correct output for a broad class of rate laws, with the class being broad enough to capture adversarial behavior. To define the broad class of rate laws for fair computation, we start with the previously defined notion of valid rate schedules, which captures a very general class of chemical kinetics. Nonetheless, we must add an additional condition. In our original definition (Definition 2.22), the reaction rate f_α can vary arbitrarily over time as long as α is applicable whenever f_α is positive. There is no requirement the other way—that a reaction must occur with positive rate if it is applicable—allowing for a greater variety of paths (e.g., segment paths with zero flux through some reactions). But, since there is nothing to prevent an adversary from “starving” reactions when they are applicable, preventing convergence, we now need to impose an additional requirement that we call fairness. We formalize this as a strictly positive lower bound H on the reaction rate at states where the reaction is applicable. In particular, while the reaction rate vector $f(t)$ is a function of the time t , $H(c)$ is a function of the state c . We allow this lower bound to be violated occasionally, so long as it holds for an infinite measure of time. (For example, a fair rate schedule could starve applicable reactions on the unit time intervals $[0, 1]$, $[2, 3]$, $[4, 5]$, \dots)

Definition 4.1. Suppose $f : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}^R$ is a valid rate schedule starting at c . We say that f is *fair* if there is a continuous function $H : \mathbb{R}_{\geq 0}^\Lambda \rightarrow \mathbb{R}_{\geq 0}^R$ such that, for all reactions $\alpha \in R$, $H_\alpha(c) > 0$ if and only if α is applicable at c , and for some subset of times $T_\alpha \subseteq \mathbb{R}_{\geq 0}$ of infinite measure, $f_\alpha(t) \geq H_\alpha(\rho(t))$ for all $t \in T_\alpha$.

Requiring that the lower bound H be continuous as a function of the state helps to ensure that if H converges to zero then the point of convergence is a static state (no reaction is applicable): By continuity, the point of convergence must have $H = \mathbf{0}$, which implies that no reaction is applicable by Definition 4.1. Note also that T_α can depend on α ; for example, Definition 4.1 allows for T_α and T_β to be disjoint for different reactions α, β (i.e., whenever we run one applicable reaction, we starve another applicable reaction).

All typically considered rate laws such as mass-action, Michaelis-Menten, Hill function kinetics, and so on, are fair. We explicitly note this for mass-action CRNs with non-divergent trajectories:

LEMMA 4.2. *The valid rate schedule for mass-action CRNs, as defined in Lemma 2.24, is fair if well-defined for all times.*

PROOF. In the notation of Lemma 2.24, take $\mathbf{H} = \mathbf{A}$. Because we always assume that the rate constants of a mass action system are positive, $\mathbf{A}_\alpha(\mathbf{c})$ is positive if and only if α is applicable at \mathbf{c} . Also, Lemma 2.24 shows that $\mathbf{f}_\alpha(t) = \mathbf{A}_\alpha(\boldsymbol{\rho}(t))$ for all $t \geq 0$, so certainly $\mathbf{f}_\alpha(t) \geq \mathbf{A}_\alpha(\boldsymbol{\rho}(t))$. \square

CRCs satisfying the definition below converge to the correct output despite actions of a very powerful adversary. Intuitively, the adversary is allowed to control the rates of all the reactions throughout the computation as long as applicable reactions are not entirely prevented from occurring.

Definition 4.3. We say a CRC *fairly computes* a function $f : \mathbb{R}_{\geq 0}^k \rightarrow \mathbb{R}_{\geq 0}$ if, for every input state \mathbf{x} , every fair rate schedule starting at \mathbf{x} with trajectory $\boldsymbol{\rho}$ obeys $\lim_{t \rightarrow \infty} (\boldsymbol{\rho}(t) \upharpoonright \{Y\}) = f(\mathbf{x})$.

There is a natural generalization of the above definition when $f : \mathbb{R}_{\geq 0}^k \rightarrow \mathbb{R}_{\geq 0}^l$ has multiple outputs: Require the trajectory $\boldsymbol{\rho}$ to obey $\lim_{t \rightarrow \infty} (\boldsymbol{\rho}(t) \upharpoonright \{Y_1, \dots, Y_l\}) = f(\mathbf{x})$, where $\{Y_1, \dots, Y_l\}$ is the set of all output species.

We now want to establish the connection between fair computation and stable computation. The following lemma shows that in one direction, the connection is immediate: Fair computation is at least as strong as stable computation.

LEMMA 4.4. *Any CRC that fairly computes a function f also stably computes f .*

PROOF. We will prove the contrapositive: Suppose that a CRC C does not stably compute f . We want to show that C does not fairly compute f , and to do this, we will find an initial state \mathbf{x} and a fair rate schedule \mathbf{f} starting at \mathbf{x} so the trajectory $\boldsymbol{\rho}$ does not obey $\lim_{t \rightarrow \infty} (\boldsymbol{\rho}(t) \upharpoonright \{Y\}) = f(\mathbf{x})$.

Since we assumed that C does not stably compute f , by Theorem 3.3, we know that there is some $\epsilon > 0$ and some input state \mathbf{x} and state \mathbf{z} so $\mathbf{x} \rightsquigarrow \mathbf{z}$ and every \mathbf{o} reachable from \mathbf{z} there is an \mathbf{o}' reachable from \mathbf{o} with $|\mathbf{o}'(Y) - f(\mathbf{x})| > \epsilon$. In other words, it is possible to reach a state \mathbf{z} , after which the output can be always made incorrect by some amount ϵ . By Lemma 2.26, we know that there is a valid rate schedule \mathbf{f}' that reaches from \mathbf{x} to \mathbf{z} in finite time t_0 . Set $\mathbf{f} = \mathbf{f}'$ on the interval $[0, t_0]$.

Now choose some positive rate constant $k_\alpha > 0$ for every reaction α , and on the interval $[t_0, t_0 + 1]$, set \mathbf{f} equal to the mass-action rate schedule corresponding to the rate constants k_α and initial state \mathbf{z} (this is a valid rate schedule by Lemma 2.24). By Theorem 2.27, we know that the state \mathbf{o} that is reached at time $t_0 + 1$ is segment reachable from \mathbf{z} , so by the definition of \mathbf{z} there must be some \mathbf{o}' with $\mathbf{o} \rightsquigarrow \mathbf{o}'$, where $|\mathbf{o}'(Y) - f(\mathbf{x})| > \epsilon$. Applying Lemma 2.26 again, we see that there is a valid rate schedule that reaches from \mathbf{o} to \mathbf{o}' by time t_1 , so set \mathbf{f} equal to this rate schedule on $[t_0 + 1, t_1]$.

Alternating a mass-action rate schedule with an adversarial rate schedule in this way, we can find a sequence of times $t_0, t_1, \dots \in \mathbb{R}_{\geq 0}$ so $\lim_{n \rightarrow \infty} t_n = \infty$ and a valid rate schedule \mathbf{f} so the associated trajectory $\boldsymbol{\rho}$ satisfies $|\boldsymbol{\rho}(t_n)(Y) - f(\mathbf{x})| > \epsilon$ for every $n \geq 1$. This proves that $\lim_{t \rightarrow \infty} (\boldsymbol{\rho}(t) \upharpoonright \{Y\}) \neq f(\mathbf{x})$. Furthermore, by construction, we know that on every interval $[t_n, t_n + 1]$ the rate schedule \mathbf{f} is equal to the mass-action rate schedule with fixed rate constants k_α . Since $\bigcup_{n \in \mathbb{N}} [t_n, t_n + 1]$ has infinite measure, Lemma 4.2 then shows that \mathbf{f} is fair. Since \mathbf{f} is a fair rate schedule failing to converge to the correct output, C does not fairly compute f . \square

It is natural to wonder if the converse of Lemma 4.4 holds, i.e., whether every CRC that stably computes f also fairly computes f . This is not true in general, but the following section shows that it is true for the CRNs we will construct.

4.1 Feedforward CRNs

As we saw before, for general CRCs it is possible that the output-stable state is always reachable but the mass-action trajectory does not converge to it, for instance, the reactions $X + X \rightarrow Y + Y$

and $Y + X \rightarrow X + X$ discussed at the start of Section 4. Thus, stable computation does not necessarily imply that the system will eventually produce the correct output. Since the mass-action trajectory defines a fair rate schedule (Lemma 4.2), the above example shows that some CRCs stably compute a function but do not fairly compute it; i.e., the converse of Lemma 4.4 does not hold for all CRCs.

In contrast to the above example, the feedforward property defined in this section allows us to bridge the definition of stable computation, defined in terms of reachability (what *could* happen), to convergence (what *will* happen), defined in terms of fair rate schedules.

Recall that a reaction $\alpha = \langle \mathbf{r}, \mathbf{p} \rangle$ produces a species S if $\mathbf{r}(S) < \mathbf{p}(S)$ and consumes S if $\mathbf{r}(S) > \mathbf{p}(S)$. We say a CRN is *feedforward* if the species can be ordered so every reaction that produces a species also consumes another species earlier in the ordering. Formally:

Definition 4.5. A CRN $C = (\Lambda, R)$ is *feedforward* if $\Lambda = \{S_1, \dots, S_n\}$ and its stoichiometry matrix satisfies $\mathbf{M}(i, j) > 0 \implies \exists(i' < i)\mathbf{M}(i', j) < 0$.

Intuitively, we want to avoid situations, as in the example above, where the output-stable state is always reachable but the trajectory does not converge to it. This can happen if the trajectory does not converge or converges to a dynamic equilibrium where reactions balance each other. In contrast, the total flux of reactions in a feedforward CRN must be bounded, because there cannot be a complete “cycle” among the species that balance consumption with production.

We start with the following simple observation: General CRNs can have reactions such as $A + B \rightarrow A + 2B + C$ that do not consume any reactant, and such CRNs clearly have infinite total flux. Luckily, by our definition of a CRN, any reaction α must either produce or consume some species, and if α produces a species, the feedforward condition guarantees that α consumes some other species.

OBSERVATION 4.6. Every reaction in a feedforward CRN consumes some species.

The following lemma will be used in formalizing the idea that a feedforward CRN cannot converge to a dynamic equilibrium (like the example beginning this section). General CRNs can have reactions that undo each other’s effect (for instance, $A \rightarrow B$ and $B \rightarrow A$). For such CRNs, we cannot bound total flux as a function of the change in species concentration—indeed, concentrations might remain constant but the two reactions canceling each other can have arbitrarily large flux—allowing for a dynamic equilibrium. In contrast, for feedforward CRNs, the following lemma shows that total flux can be bounded by the change in state:

LEMMA 4.7. For a feedforward CRN, for each reaction α , there is a constant K_α independent of \mathbf{u} such that $\|\mathbf{Mu}\| < \epsilon$ implies that $\mathbf{u}_\alpha < K_\alpha \epsilon$.

PROOF. We show by induction that we can find such a K_β for every reaction β that consumes a species. First suppose β is a reaction that consumes the first species S_1 in the feedforward ordering. By the feedforward property, no reaction can produce S_1 , so for each reaction α , we have $\mathbf{M}(S_1, \alpha) \leq 0$. Thus,

$$|(\mathbf{Mu})(S_1)| = \sum_{\alpha \in R} |\mathbf{M}(S_1, \alpha)| \mathbf{u}_\alpha.$$

Since the left-hand side of this equation is less than ϵ , and the right-hand side is a sum of non-negative terms, each term on the right-hand side must be less than ϵ . Since β consumes S_1 , $|\mathbf{M}(S_1, \beta)| \geq 1$, so $\mathbf{u}_\beta < \epsilon$ and we can take $K_\beta = 1$. This establishes the base case for S_1 .

Now assume inductively that we have found an appropriate constant K_α for every reaction α that consumes a species S_i for $i < n$, and suppose that β consumes S_n . Then, because

$|(\mathbf{M}\mathbf{u})(S_n)| < \epsilon$, we know that

$$\sum_{\substack{\alpha \in R \\ \mathbf{M}(S_n, \alpha) < 0}} |\mathbf{M}(S_n, \alpha)| \mathbf{u}_\alpha < \epsilon + \sum_{\substack{\alpha \in R \\ \mathbf{M}(S_n, \alpha) > 0}} |\mathbf{M}(S_n, \alpha)| \mathbf{u}_\alpha < \epsilon \cdot \left[1 + \sum_{\substack{\alpha \in R \\ \mathbf{M}(S_n, \alpha) > 0}} |\mathbf{M}(S_n, \alpha)| K_\alpha \right],$$

where for the second inequality we have used the feedforward condition to conclude that every reaction producing S_n must consume S_i for some $i < n$, and therefore have flux bounded by $K_\alpha \epsilon$ by inductive assumption. Like before, the leftmost term in the inequality is a sum of non-negative terms, so if we take

$$K_\beta = 1 + \sum_{\substack{\alpha \in R \\ \mathbf{M}(\alpha, S_n) > 0}} |\mathbf{M}(S_n, \alpha)| K_\alpha,$$

then $\mathbf{u}_\beta < K_\beta \epsilon$. This shows that an appropriate K_β exists for every reaction β that consumes a species. But by Observation 4.6, every reaction consumes a species, so we are done. \square

Finally, we are ready to prove the main lemma about feedforward CRNs, a consequence of which (Lemma 4.9) establishes the connection between stable computation and convergence to the output-stable state for feedforward CRNs.

LEMMA 4.8. *Consider a feedforward CRN, and suppose \mathbf{f} is a valid rate schedule. Then the corresponding trajectory $\boldsymbol{\rho}$ converges to a state \mathbf{y} in the limit time $t \rightarrow \infty$. Further, if \mathbf{f} is fair, then \mathbf{y} is static, i.e., no reaction is applicable in \mathbf{y} .*

PROOF. First, define

$$K = 1 + \max_{\alpha \in R} \sum_{S_i \in \Lambda} \max(0, \mathbf{M}(S_i, \alpha)).$$

In particular, K is larger than the sums of the positive entries in the columns of \mathbf{M} . Intuitively, we will assign a “mass” to each species, and K will be the ratio of the masses assigned to consecutive species in the feedforward ordering. By making K sufficiently large, we can guarantee that running any reaction has the effect of decreasing the total mass $V(\mathbf{x})$ of any state \mathbf{x} . Formally, define $V : \mathbb{R}_{\geq 0}^\Lambda \rightarrow \mathbb{R}_{\geq 0}$ as

$$V(\mathbf{x}) = \sum_{i=1}^n \frac{\mathbf{x}(S_i)}{K^i}.$$

Recall $\mathbf{f}_\alpha(t)$ is the rate of reaction α at time t . Note that whenever $\boldsymbol{\rho}(t)$ is differentiable²⁰

$$\begin{aligned} \frac{d}{dt} V(\boldsymbol{\rho}(t)) &= \sum_{i=1}^n \frac{d}{dt} \frac{(\boldsymbol{\rho}(t))(S_i)}{K^i} \\ &= \sum_{i=1}^n \frac{1}{K^i} \sum_{\alpha \in R} \mathbf{M}(S_i, \alpha) \mathbf{f}_\alpha(t) \\ &= \sum_{\alpha \in R} \sum_{i=1}^n \frac{1}{K^i} \mathbf{M}(S_i, \alpha) \mathbf{f}_\alpha(t) \\ &= \sum_{\alpha \in R} C_\alpha \mathbf{f}_\alpha(t), \end{aligned}$$

²⁰By Lebesgue’s fundamental theorem of calculus [43, Theorem 6.11, Theorem 6.14], applied to $\mathbf{f}_\alpha(t)$, we know that $\boldsymbol{\rho}(t)$ is locally absolutely continuous, and almost everywhere differentiable with derivative $\mathbf{M} \cdot \mathbf{f}(t)$.

where $C_\alpha = \sum_{i=1}^n \frac{1}{K^i} \mathbf{M}(S_i, \alpha)$. For any fixed α , let i_0 be the smallest i such that $\mathbf{M}(S_i, \alpha) \neq 0$, i.e., S_{i_0} is the first species in the feedforward ordering that is produced or consumed by α . By the feedforward condition, $\mathbf{M}(S_{i_0}, \alpha) \leq -1$, i.e., S_{i_0} is consumed. As a result,

$$\begin{aligned} C_\alpha &= \frac{1}{K^{i_0}} \left(\mathbf{M}(S_{i_0}, \alpha) + \frac{1}{K} \sum_{i=i_0+1}^n \frac{\mathbf{M}(S_i, \alpha)}{K^{i-i_0-1}} \right) \\ &< \frac{1}{K^{i_0}} (\mathbf{M}(S_{i_0}, \alpha) + 1) \\ &\leq 0. \end{aligned}$$

Now, we show that the total flux through all of the reactions is finite. To fix notation, let us write \mathbf{F}_α for the total flux through reaction α as $t \rightarrow \infty$, i.e.,

$$\mathbf{F}_\alpha = \int_0^\infty \mathbf{f}_\alpha(t) dt.$$

Since every C_α is negative, for any fixed reaction α_0 ,

$$\frac{d}{dt} V(\boldsymbol{\rho}(t)) = \sum_{\alpha \in R} C_\alpha \mathbf{f}_\alpha(t) \leq C_{\alpha_0} \mathbf{f}_{\alpha_0}(t)$$

so by integrating²¹ both sides

$$\lim_{t \rightarrow \infty} V(\boldsymbol{\rho}(t)) - V(\mathbf{c}) \leq C_{\alpha_0} \int_0^\infty \mathbf{f}_{\alpha_0}(t) dt = C_{\alpha_0} \mathbf{F}_{\alpha_0}.$$

If \mathbf{F}_{α_0} were infinite, then because $C_{\alpha_0} < 0$, we see that $V(\boldsymbol{\rho}(t))$ would be unbounded below. Since $\boldsymbol{\rho}(t)$ always remains in $\mathbb{R}_{\geq 0}^\Lambda$ where $V(\boldsymbol{\rho}(t)) \geq 0$, we conclude that \mathbf{F}_{α_0} must be finite.

If we write $\mathbf{v}_\alpha \in \mathbb{R}^\Lambda$ for the α -column of \mathbf{M} (i.e., $\mathbf{v}_\alpha(S) = \mathbf{M}(S, \alpha)$), then

$$\begin{aligned} \int_0^\infty \left\| \frac{d\boldsymbol{\rho}(t)}{dt} \right\| dt &= \int_0^\infty \left\| \sum_{\alpha \in R} \mathbf{v}_\alpha \mathbf{f}_\alpha(t) \right\| dt \\ &\leq \int_0^\infty \sum_{\alpha \in R} \mathbf{f}_\alpha(t) |\mathbf{v}_\alpha| dt \\ &= \sum_{\alpha \in R} |\mathbf{v}_\alpha| \int_0^\infty \mathbf{f}_\alpha(t) dt = \sum_{\alpha \in R} |\mathbf{v}_\alpha| \mathbf{F}_\alpha \\ &< \infty. \end{aligned}$$

In other words, $\boldsymbol{\rho}$ has finite length.

Let us now show that $\boldsymbol{\rho}$ having finite length implies that it converges. For any states \mathbf{x}, \mathbf{z} define $d(\mathbf{x}, \mathbf{z}) = \|\mathbf{x} - \mathbf{z}\|$ as the Euclidean distance from \mathbf{x} to \mathbf{z} . It suffices to show

$$\text{For all } \epsilon > 0 \text{ there is some } M \in \mathbb{R}_{\geq 0} \text{ such that for all } t, s > M, d(\boldsymbol{\rho}(t), \boldsymbol{\rho}(s)) < \epsilon. \quad (*)$$

Indeed, taking this as given, let $\mathbf{x}_k = \boldsymbol{\rho}(k)$. Then the sequence $(\mathbf{x}_k)_{k=1}^\infty$ is a Cauchy sequence in \mathbb{R}^Λ , so it must converge to some state \mathbf{y} . Moreover, we actually know that $\boldsymbol{\rho}(t)$ converges to some state \mathbf{y} as $t \rightarrow \infty$: For any $\epsilon > 0$ there is some N such that for all $n > N$, we know $d(\mathbf{x}_n, \mathbf{y}) < \epsilon/2$

²¹Note that $\int_0^\infty \frac{d}{dt} V(\boldsymbol{\rho}(t)) dt = \lim_{t \rightarrow \infty} V(\boldsymbol{\rho}(t)) - V(\mathbf{c})$ by the fundamental theorem of calculus [43, Theorem 6.10]. To use the fundamental theorem of calculus, we must ensure that $V(\boldsymbol{\rho}(t))$ is locally absolutely continuous—this follows from the fact that the trajectory $\boldsymbol{\rho}(t)$ is defined in terms of a (Lebesgue) integral.

and there is some M such that for all $t, s > M$, we know $d(\rho(t), \rho(s)) < \epsilon/2$. In particular, taking m to be an integer larger than N and M , we see that

$$d(\rho(t), y) \leq d(\rho(t), x_m) + d(x_m, y) < \epsilon$$

for any $t > M$. This shows that, taking (*) for granted, ρ must converge to y .

Let us now prove (*). Proceed by contradiction, so suppose that there is some $\epsilon > 0$ such that for all $M \in \mathbb{R}_{\geq 0}$, there exists some $t > s > M$ such that $d(\rho(t), \rho(s)) \geq \epsilon$. Then take $M_1 = 0$, and label the points we get by this assumption t_1, s_1 . Then for any $n > 1$, take $M_n = t_{n-1}$ and label the next pair of points t_n, s_n . Then

$$\begin{aligned} \int_0^\infty \left\| \frac{d\rho(t)}{dt} \right\| dt &\geq \sum_{i=1}^\infty \int_{s_i}^{t_i} \left\| \frac{d\rho(t)}{dt} \right\| dt \\ &\geq \sum_{i=1}^\infty d(\rho(t_i), \rho(s_i)) \\ &\geq \sum_{i=1}^\infty \epsilon = \infty, \end{aligned}$$

where the second inequality uses the absolute continuity of ρ . This gives a contradiction, establishing (*), which proves that $\rho(t)$ converges to some state y as $t \rightarrow \infty$.

Now let us assume that f is fair and establish that y is a static state. If not, then there is some reaction α applicable at y , so $H_\alpha(y) = C > 0$. By the continuity of H_α , there is some δ so $H_\alpha(x) > C/2$ for all $x \in \mathbb{R}_{\geq 0}^\Lambda$ with $\|x - y\| < \delta$. Since ρ converges to y , this implies that $H_\alpha(\rho(t)) > C/2$ for all t greater than some t_0 .

For any measurable $U \subseteq \mathbb{R}_{\geq 0}$, let $\mu(U) \in \mathbb{R}_{\geq 0} \cup \{\infty\}$ denote the measure of U . Letting T_α be the subset of $\mathbb{R}_{\geq 0}$ as in Definition 4.1 where $f_\alpha(t) \geq H_\alpha(\rho(t))$ for $t \in T_\alpha$ and $\mu(T_\alpha) = \infty$, we see that $f_\alpha(t) > C/2$ for every $t \in T_\alpha \cap [t_0, \infty)$. Also note that $\mu(T_\alpha \cap [t_0, \infty)) = \infty$. By the contrapositive direction of Lemma 4.7, we see that this implies that $\| \frac{d\rho(t)}{dt} \| > C/2K_\alpha$ for every $t \in T_\alpha \cap [t_0, \infty)$. As a result,

$$\int_0^\infty \left\| \frac{d\rho(t)}{dt} \right\| dt \geq \int_{T_\alpha \cap [t_0, \infty)} \left\| \frac{d\rho(t)}{dt} \right\| dt \geq \mu(T_\alpha \cap [t_0, \infty)) \frac{C}{2K_\alpha} = \infty,$$

contradicting the finite length of ρ . □

By applying Lemma 4.8 to a feedforward CRN that also stably computes a function, we obtain the following result, which states that such a CRN will reach the correct output under any fair rate schedule, from any state reachable from x . This is almost a converse to Lemma 4.4; however, note that unlike in Lemma 4.4, the CRC is required to be feedforward.

LEMMA 4.9. *Any feedforward CRC that stably computes a function f also fairly computes f .*

PROOF. By Lemma 4.8, under any fair rate schedule, the CRC converges from the initial state x to a static state y . By Theorem 2.27, since y is reachable from x under a valid rate schedule, we know $x \rightsquigarrow y$. Since y is static, it is output-stable. Then, because the CRC stably computes f and $x \rightsquigarrow y$, we must have $y(Y) = f(x)$ or else it would have stabilized to an incorrect output. So, the CRC fairly computes f . □

We point out that the feedforward property is not a *necessary* condition for stable computation to coincide with fair computation. For example, consider the non-feedforward CRC with reactions $X + X \rightarrow R + Y$ and $R + R \rightarrow X$. This CRC stably computes $f(x) = (2/3)x$ [52], and it can also be

shown that it fairly computes f . The key property it shares with feedforward CRCs is Lemma 4.7: A large flux through its reactions implies a large change in state.

LEMMA 4.10. *If the CRC is feedforward, then the mass-action trajectory $\rho : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}^\Lambda$ as defined in Lemma 2.24 is defined at all times.*

PROOF. By the Escape Lemma [36, Lemma 9.19] it suffices to show that ρ remains in a compact subset of $\mathbb{R}_{\geq 0}^\Lambda$ for all times when it is defined. Let $V(\mathbf{x})$ be as in the proof of Lemma 4.8. We know by the argument of Lemma 4.8 that $\frac{d}{dt}V(\rho(t)) \leq 0$, so for all t , we know that $\rho(t)$ lies in the subset of $\mathbb{R}_{\geq 0}^\Lambda$ where $V(\mathbf{x}) \leq V(\rho(0))$. Since V is a linear function with strictly positive coefficients, this is a compact subset of $\mathbb{R}_{\geq 0}^\Lambda$. \square

Since mass-action yields a valid rate schedule (Lemma 2.24) that is fair (Lemma 4.2) and defined at all times for feedforward CRNs (Lemma 4.10), the following corollary is immediate. The corollary states that for a CRC stably computing a function f , the CRC will also converge to the correct output under mass-action kinetics, no matter the positive rate constants, and even if an adversary can first “steer” the CRC to some reachable state before letting mass-action kinetics take over.

COROLLARY 4.11. *Consider a feedforward CRC stably computing a function f . Then for any input state \mathbf{x} , for any state \mathbf{z} reachable from \mathbf{x} , for any choice of reaction rate constants, the mass-action trajectory starting at \mathbf{z} is defined for all times and converges to an output-stable state \mathbf{y} in the limit $t \rightarrow \infty$ such that $\mathbf{y}(Y) = f(\mathbf{x})$.*

5 THE COMPUTATIONAL POWER OF STABLE COMPUTATION

This section presents the main results of our article, delineating the computational power of stable computation. As justified in Sections 3 and 4, we use stable computation as our primary notion of rate-independence. In Section 5.1, we discuss the input and output representation of negative quantities and the composition of CRN modules via the “dual-rail representation.” In Section 5.2, we summarize our results on the computational power of stable computation for direct and dual-rail representations. These results are proven in subsequent sections, with positive (Section 5.3 for dual-rail, Section 5.4 for direct) and negative directions (Section 5.4 for direct, Section 5.6 for dual-rail) separately.

5.1 Dual-rail Representations

The *direct* concentration-to-value mapping articulated in Definition 3.2 is a straightforward way to represent non-negative input and output values. However, there are two reasons why an alternative, albeit more complex, encoding may be preferred. First, since concentrations cannot be negative, computable functions are restricted to non-negative domain and range. Second, as explained below, the direct output encoding frustrates the composition of smaller CRC modules into larger CRCs. The *dual-rail* representation we introduce in this section is a natural way to solve both problems.

A natural way to represent a (possibly negative) real value in chemistry is to encode it as the difference of two concentrations. Formally, let $f : \mathbb{R}^k \rightarrow \mathbb{R}$ be a function. A function $\hat{f} : \mathbb{R}_{\geq 0}^{2k} \rightarrow \mathbb{R}_{\geq 0}^2$ is a *dual-rail representation* of f if, for all $\mathbf{x}^+, \mathbf{x}^- \in \mathbb{R}_{\geq 0}^k$, if $(y^+, y^-) = \hat{f}(\mathbf{x}^+, \mathbf{x}^-)$, then $f(\mathbf{x}^+ - \mathbf{x}^-) = y^+ - y^-$. In other words, \hat{f} represents f as the difference of its two outputs y^+ and y^- , and it works for any input pair $(\mathbf{x}^+, \mathbf{x}^-)$ whose difference is the input value to f . We can define a CRC to stably compute such a function in the same manner as in Section 3, but having $2k$ input species $\Sigma = \{X_1^+, X_1^-, X_2^+, X_2^-, \dots, X_k^+, X_k^-\}$ and two output species $\Gamma = \{Y^+, Y^-\}$.

Definition 5.1. We say that a CRC *stably dual-computes* $f : \mathbb{R}^k \rightarrow \mathbb{R}$ if it stably computes a dual-rail representation $\hat{f} : \mathbb{R}_{\geq 0}^k \times \mathbb{R}_{\geq 0}^k \rightarrow \mathbb{R}_{\geq 0} \times \mathbb{R}_{\geq 0}$ of f .

This definition implies that, for all $\mathbf{x} = (\mathbf{x}^+, \mathbf{x}^-) \in \mathbb{R}_{\geq 0}^{2k}$, for all \mathbf{c} such that $\mathbf{x} \rightsquigarrow \mathbf{c}$, there exists an output-stable state \mathbf{o} such that $\mathbf{c} \rightsquigarrow \mathbf{o}$ and $\mathbf{o}(Y^+) - \mathbf{o}(Y^-) = f(\mathbf{x}^+ - \mathbf{x}^-)$. Note that a single function has an infinite number of dual-rail representations; we require only that a CRC exists to compute one of them to say that the function is stably dual-computable by a CRC.

Besides making negative values chemically representable, we will see that the dual-rail representation plays a key role in allowing the composition of smaller CRC modules into a larger CRC. A key concept to enable such composition is *output-obliviousness*:

Definition 5.2. A CRC $C = (\Lambda, R, \Sigma, \Gamma)$ is *output-oblivious* if none of its output species Γ is a reactant in any reaction. In other words, for every $\alpha = \langle \mathbf{r}, \mathbf{p} \rangle \in R$ and $Y \in \Gamma$, $\mathbf{r}(Y) = 0$.

To recognize the problem with composition of the direct representation (Definition 3.2), define the *composition* of two CRCs as the CRC that has the union of their reactions, relabeling the output species of the upstream CRN to be the input species of the downstream one [17, Definition 16]. Then with the direct output representation, output-obliviousness is necessary for composability but provably restricts computational power [17]: (1) Composing two stably computing CRCs stably computes the function composition if and only if the upstream CRC is output-oblivious (except in trivial cases). Intuitively, the downstream CRC can interfere with the upstream computation by prematurely consuming the output. (2) An output-oblivious CRC can only stably compute “super-additive” functions. For example, any CRC stably computing the function $f(x_1, x_2) = x_1 - x_2$ must necessarily be able to consume its output species, since a state with more than the desired amount of output is reachable by additivity (e.g., state with x_1 amount of Y).

Our results imply that the dual-rail representation allows composition without sacrificing computational power. In particular, our dual-rail constructions are all output-oblivious and thus composable by concatenation, and our negative results apply to dual-rail CRCs whether or not they are output-oblivious. To see roughly why the dual-rail representation helps with composition, consider the $f(x_1, x_2) = x_1 - x_2$ function above. We can now compute this function with an output-oblivious (and therefore composable) CRC by *producing* Y^- to decrease the value of the output, without consuming any output species.

Recall fair computation (Definition 4.3). We define dual-rail fair computation analogously.

Definition 5.3. We say a CRC *fairly dual-computes* a function $f : \mathbb{R}^k \rightarrow \mathbb{R}$ if it fairly computes a dual-rail representation of f .

Since dual-rail computation is defined by a CRC that directly computes a dual-rail representation function, Lemmas 4.4 and 4.9 imply the analogous results for dual-rail:

LEMMA 5.4. *If a CRC fairly dual-computes a function $f : \mathbb{R}^k \rightarrow \mathbb{R}$, then it stably dual-computes f .*

PROOF. The proof is similar to the proof of Lemma 4.4, except one uses Corollary 3.4 in the place of Theorem 3.3. \square

LEMMA 5.5. *If a feedforward CRC stably dual-computes a function $f : \mathbb{R}^k \rightarrow \mathbb{R}$, then it fairly dual-computes f .*

PROOF. The proof is the same as Lemma 4.9, since a static state is output-stable regardless of how many output species there are. \square

5.2 Statement of Main Results

Below, we summarize our results about the computational power of stable computation, which we prove in subsequent subsections. First, we formally define the relevant classes of functions that will correspond to direct and dual-rail stable computation:

Definition 5.6. A function $f : \mathbb{R}^k \rightarrow \mathbb{R}$ is *rational linear* if there exist $a_1, \dots, a_k \in \mathbb{Q}$ such that $f(\mathbf{x}) = \sum_{i=1}^k a_i x(i)$. A function $f : \mathbb{R}^k \rightarrow \mathbb{R}$ is *rational affine* if there exist $a_1, \dots, a_k, c \in \mathbb{Q}$ such that $f(\mathbf{x}) = \sum_{i=1}^k a_i x(i) + c$, i.e., f is a rational constant c plus a rational linear function.

We note that rational linearity has the equivalent characterization that f is linear and maps rational inputs $\mathbf{x} \in \mathbb{Q}^n$ to rational outputs.

Definition 5.7. A function $f : \mathbb{R}^k \rightarrow \mathbb{R}$ is *piecewise rational linear (affine)* if there is a finite set of partial rational linear (affine) functions $f_1, \dots, f_p : \mathbb{R}^k \rightarrow \mathbb{R}$, with $\bigcup_{j=1}^p \text{dom } f_j = \mathbb{R}^k$, such that, for all $j \in \{1, \dots, p\}$ and all $\mathbf{x} \in \text{dom } f_j$, $f(\mathbf{x}) = f_j(\mathbf{x})$. In this case, we say that f_1, \dots, f_p are the *components* of f .

Definition 5.8. A function $f : \mathbb{R}_{\geq 0}^k \rightarrow \mathbb{R}_{\geq 0}$ is *positive-continuous* if, for all $U \subseteq \{1, \dots, k\}$, f is continuous on the domain

$$D_U = \left\{ \mathbf{x} \in \mathbb{R}_{\geq 0}^k \mid (\forall i \in \{1, \dots, k\}) x(i) > 0 \iff i \in U \right\}.$$

In other words, f is continuous on any subset $D \subset \mathbb{R}_{\geq 0}^k$ that does not have any coordinate $i \in \{1, \dots, k\}$ that takes both zero and positive values in D .

The following theorems are the main results of this article, exactly characterizing the functions stably computable with direct and dual-rail representation of inputs and outputs, and showing the equivalence between fair computation and stable computation. Furthermore, although non-feedforward CRCs do exist to compute functions in this class, the theorem shows that feedforward CRCs suffice to compute all such functions.

THEOREM 5.9. *For a function $f : \mathbb{R}_{\geq 0}^k \rightarrow \mathbb{R}_{\geq 0}$, the following are equivalent:*

- (1) *f is fairly computable by a CRC.*
- (2) *f is stably computable by a CRC.*
- (3) *f is positive-continuous piecewise rational linear.*
- (4) *f is stably computable by a feedforward CRC.*
- (5) *f is fairly computable by a feedforward CRC.*

PROOF. (1) implies (2) is Lemma 4.4. (2) implies (3) is Lemma 5.38. (3) implies (4) is Lemma 5.16. (4) implies (5) is Lemma 4.9. (5) implies (1) is obvious. \square

The following is the dual-rail analog of Theorem 5.9.

THEOREM 5.10. *For a function $f : \mathbb{R}^k \rightarrow \mathbb{R}$, the following are equivalent:*

- (1) *f is fairly dual-computable by a CRC.*
- (2) *f is stably dual-computable by a CRC.*
- (3) *f is continuous piecewise rational linear.*
- (4) *f is stably dual-computable by a feedforward, output-oblivious CRC.*
- (5) *f is fairly dual-computable by a feedforward, output-oblivious CRC.*

PROOF. (1) implies (2) is Lemma 5.4. (2) implies (3) is Lemma 5.40. (3) implies (4) is Lemma 5.15. (4) implies (5) is Lemma 5.5. (5) implies (1) is obvious. \square

5.3 Positive Result: Continuous Piecewise Rational Linear Functions are Dual-rail Computable

Definition 5.7 does not stipulate how complex the “boundaries” between the linear pieces of a piecewise rational linear function can be. The boundaries can even be irrational in some sense, e.g., the function $f(x_1, x_2) = 0$ if $x_1 > \sqrt{2} \cdot x_2$ and $f(x_1, x_2) = x_1 + x_2$ otherwise. However, if we additionally require that f be *continuous*, then the following theorem of Ovchinnikov [42, Theorem 2.1] states that f has a particularly clean form, conducive to computation by CRCs.

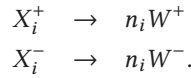
THEOREM 5.11 ([42], THEOREM 2.1). *Let $D \subseteq \mathbb{R}^k$ be convex. For every continuous piecewise affine function $f : D \rightarrow \mathbb{R}$ with components g_1, \dots, g_p , there exists a family $S_1, \dots, S_q \subseteq \{1, \dots, p\}$ such that, for all $\mathbf{x} \in D$, $f(\mathbf{x}) = \max_{i \in \{1, \dots, q\}} \min_{j \in S_i} g_j(\mathbf{x})$.*

Note that as a special case, the above result applies when f is continuous piecewise rational linear. The above theorem as stated slightly generalizes the result due to Ovchinnikov [42] (by not requiring D to be closed), although the proof technique is essentially the same. For completeness, we provide the proof in Appendix B.

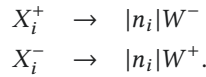
We use the theorem above to dual-compute continuous piecewise rational linear functions by composing CRC modules for rational linear functions, min, and max. These modules are developed in the following three lemmas:

LEMMA 5.12. *Rational linear functions are stably dual-computable by a feedforward, output-oblivious CRC.*

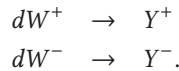
PROOF. Let $g : \mathbb{R}^k \rightarrow \mathbb{R}$ be a rational linear function $g(\mathbf{x}) = \sum_{i=1}^k a_i x(i)$. By clearing denominators, there exist $n_1, \dots, n_k \in \mathbb{Z}$ and $d \in \mathbb{Z}^+$ such that $g(\mathbf{x}) = \frac{1}{d} \sum_{i=1}^k n_i x(i)$. The following reactions compute a dual-rail representation of g with input species $X_1^+, \dots, X_k^+, X_1^-, \dots, X_k^-$ and output species Y^+, Y^- . For each i such that $n_i > 0$, add the reactions



For each i such that $n_i < 0$, add the reactions



To divide the values of W^- and W^+ by d , add the reactions



In particular, these reactions compute the dual-rail representation $\hat{g} : \mathbb{R}_{\geq 0}^{2k} \rightarrow \mathbb{R}_{\geq 0}^2$ where

$$\hat{g}(x_1^+, \dots, x_k^+, x_1^-, \dots, x_k^-) = (y^+, y^-) = \left(\frac{1}{d} \left[\sum_{n_i > 0} n_i x_i^+ + \sum_{n_i < 0} |n_i| x_i^- \right], \frac{1}{d} \left[\sum_{n_i > 0} n_i x_i^- + \sum_{n_i < 0} |n_i| x_i^+ \right] \right).$$

It is straightforward to verify that \hat{g} really is a dual-rail representation of g . To see that the above CRC stably computes \hat{g} , define the functions $p, q : \mathbb{R}_{\geq 0}^\Lambda \rightarrow \mathbb{R}$ so

$$\begin{aligned} p(\mathbf{c}) &= \mathbf{c}(Y^+) + \frac{1}{d} \mathbf{c}(W^+) + \frac{1}{d} \sum_{n_i > 0} n_i \mathbf{c}(X_i^+) + \frac{1}{d} \sum_{n_i < 0} |n_i| \mathbf{c}(X_i^-) \\ q(\mathbf{c}) &= \mathbf{c}(Y^-) + \frac{1}{d} \mathbf{c}(W^-) + \frac{1}{d} \sum_{n_i > 0} n_i \mathbf{c}(X_i^-) + \frac{1}{d} \sum_{n_i < 0} |n_i| \mathbf{c}(X_i^+). \end{aligned}$$

It is also straightforward to verify that both p and q are preserved by all of the reactions in the above CRC. This shows that for all \mathbf{c} and \mathbf{d} with $\mathbf{c} \rightsquigarrow \mathbf{d}$, we have $p(\mathbf{d}) = p(\mathbf{c})$ and $q(\mathbf{d}) = q(\mathbf{c})$. Now observe that from any state it is always possible to reach a state \mathbf{o} that only has positive concentrations of Y^+ and Y^- by executing the reactions above to completion in the order in which they are listed. Such a state is evidently output-stable, so for any input state \mathbf{x} and any \mathbf{c} reachable from \mathbf{x} , there is an output-stable state \mathbf{o} reachable from \mathbf{c} . Furthermore, since \mathbf{o} only has positive concentrations of Y^+ and Y^- , we know that $p(\mathbf{o}) = \mathbf{o}(Y^+)$ and $q(\mathbf{o}) = \mathbf{o}(Y^-)$. Since \mathbf{o} is reachable from \mathbf{x} , we see that

$$\begin{aligned} \mathbf{o}(Y^+) &= p(\mathbf{o}) = p(\mathbf{x}) = \frac{1}{d} \left[\sum_{n_i > 0} n_i x_i^+ + \sum_{n_i < 0} |n_i| x_i^- \right] \\ \mathbf{o}(Y^-) &= q(\mathbf{o}) = q(\mathbf{x}) = \frac{1}{d} \left[\sum_{n_i > 0} n_i x_i^- + \sum_{n_i < 0} |n_i| x_i^+ \right]. \end{aligned}$$

This shows that a dual-rail representation of any rational linear function can be stably dual-computed by a CRC. Moreover, the CRC above is clearly output-oblivious, and it is feedforward under the ordering $X_1^+ < \dots < X_n^+ < X_1^- < \dots < X_n^- < W^+ < W^- < Y^+ < Y^-$. \square

LEMMA 5.13. *Min is stably dual-computable by a feedforward, output-oblivious CRC.*

PROOF. The following reactions stably compute a dual-rail representation of min with input species $X_1^+, X_2^+, X_1^-, X_2^-$ and output species Y^+, Y^- .



In particular this CRC computes the dual rail representation $\hat{f} : \mathbb{R}_{\geq 0}^4 \rightarrow \mathbb{R}_{\geq 0}^2$ where

$$\hat{f}(x_1^+, x_2^+, x_1^-, x_2^-) = (y^+, y^-) = (\min(x_1^+ + x_2^-, x_2^+ + x_1^-), x_1^- + x_2^-).$$

It is straightforward to verify that \hat{f} is really a dual-rail representation of min. To see that the above CRC stably computes \hat{f} ,²² define the functions $p, q, \delta : \mathbb{R}_{\geq 0}^4 \rightarrow \mathbb{R}$ so

$$p(\mathbf{c}) = 2\mathbf{c}(Y^+) + \mathbf{c}(X_1^+) + \mathbf{c}(X_2^+) + \mathbf{c}(X_1^-) + \mathbf{c}(X_2^-)$$

$$q(\mathbf{c}) = \mathbf{c}(Y^-) + \mathbf{c}(X_1^-) + \mathbf{c}(X_2^-)$$

$$\delta(\mathbf{c}) = \mathbf{c}(X_1^+) - \mathbf{c}(X_1^-) - \mathbf{c}(X_2^+) + \mathbf{c}(X_2^-).$$

It is also straightforward to verify that the above three functions are preserved by all of the reactions in the given CRC. Note that by running 5.3, then 5.2, then 5.1 to completion it is always possible to reach a state \mathbf{o} with $\mathbf{o}(X_1^-) = \mathbf{o}(X_2^-) = 0$ and also $\mathbf{o}(X_1^+) = 0$ or $\mathbf{o}(X_2^+) = 0$. Such a state is evidently output-stable, so for any input state \mathbf{x} and any \mathbf{c} reachable from \mathbf{x} there is an output-stable state \mathbf{o} reachable from \mathbf{c} . Since \mathbf{o} is reachable from \mathbf{x} , we know that

$$\mathbf{o}(Y^-) = q(\mathbf{o}) = q(\mathbf{x}) = x_1^- + x_2^-.$$

²²This analysis of the CRC for the min function here is directly based on the definition of stable computation. Recently a powerful framework has been developed [52], based on a wide class of so-called *noncompetitive* CRCs in which no species consumed in a reaction is a reactant in another reaction (not even as a non-consumed catalyst). For such CRCs, the task of proving correctness of stable computation is greatly simplified. Since the CRC computing min is noncompetitive, that framework could be applied here to yield a simpler proof of correctness. We use our direct proof here for the sake of making the current article self-contained.

Now suppose without loss of generality that $x_1^+ + x_2^- \leq x_2^+ + x_1^-$ (the analysis of the other case is similar). Then $\delta(\mathbf{x}) \leq 0$, so $\mathbf{o}(X_1^+) - \mathbf{o}(X_2^+) = \delta(\mathbf{o}) \leq 0$. If $\mathbf{o}(X_1^+)$ were positive, then by the definition of \mathbf{o} , we would know that $\mathbf{o}(X_2^+) = 0$, contradicting the fact that $\delta(\mathbf{o}) \leq 0$. Thus, $\mathbf{o}(X_1^+) = 0$ and

$$\mathbf{o}(X_2^+) = -\delta(\mathbf{o}) = -\delta(\mathbf{x}) = x_1^- + x_2^+ - x_1^+ - x_2^-.$$

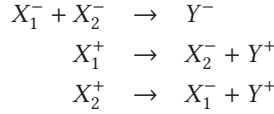
Finally, note that

$$p(\mathbf{x}) = x_1^+ + x_2^+ + x_1^- + x_2^- = p(\mathbf{o}) = 2\mathbf{o}(Y^+) + \mathbf{o}(X_2^+) = 2\mathbf{o}(Y^+) + x_1^- + x_2^+ - x_1^+ - x_2^-.$$

Solving the above equation for $\mathbf{o}(Y^+)$ shows that $\mathbf{o}(Y^+) = x_1^+ + x_2^- = \min(x_1^+ + x_2^-, x_2^+ + x_1^-)$. This shows that the above CRC stably computes \hat{f} . The CRN is clearly output-oblivious and it is feedforward with the ordering $X_1^- < X_2^- < X_1^+ < X_2^+ < Y^- < Y^+$. \square

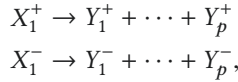
COROLLARY 5.14. *Max is stably dual-computable by a feedforward, output-oblivious CRC.*

PROOF. To stably compute a dual-rail representation of max, observe that it is equivalent to computing the min function with the roles of the “plus” and “minus” species reversed (which negates the value represented in dual-rail), because $\max(x_1, x_2) = -\min(-x_1, -x_2)$. In other words, use the reactions



LEMMA 5.15. *Let $D \subseteq \mathbb{R}^k$ be convex, and let $f : D \rightarrow \mathbb{R}$ be a continuous piecewise rational linear function. Then f is stably dual-computed by a feedforward, output-oblivious CRC.*

PROOF. By Theorem 5.11, we know that any such continuous piecewise rational linear function can be represented as a composition of max, min, and rational linear functions. Moreover, the min and max functions with two arguments can be composed in a tree of depth $\log l$ to compute the minimum or maximum functions with input arity l . Since multiple rational linear functions may use the same inputs, we also need a fan-out module (likewise feedforward, output-oblivious) that copies a single input to multiple outputs:



where p is the number of linear components of the piecewise linear function f . Because output-oblivious CRCs are composable, and because the composition of feedforward, output-oblivious CRCs is again feedforward and output-oblivious, we can compose the CRCs from Lemma 5.12, Lemma 5.13, and Corollary 5.14 to produce a feedforward, output-oblivious CRC that computes f . \square

Note that Lemma 5.15 applies for arbitrary convex domains $D \subseteq \mathbb{R}^k$, which will be useful in proving Lemma 5.16, where we take D to be a strict convex subset of $\mathbb{R}_{\geq 0}^k$ in which no coordinate takes both 0 and positive values in D . Since \mathbb{R}^k is itself convex, it also establishes the “(3) implies (4)” implication in the proof of Theorem 5.10.

5.4 Positive Result: Positive-continuous Piecewise Rational Linear Functions Are Directly Computable

The following lemma is the *direct* stable computation analog of Lemma 5.15. Intuitively, it is proven by using Lemma 5.15 to stably dual-compute 2^k different continuous piecewise rational linear

functions in parallel, one for each possible choice of which input species X_1, \dots, X_k are 0. A separate computation determines which inputs are positive and selects the appropriate output. Note that positive inputs may be discovered “piecemeal,” so the system must be robust to a continual updating of the decision. However, there is a monotonicity to this process that will make consistent updating possible: Once an input species is discovered to be present, we know for sure it is. (Whereas a species that appears to be absent simply may have not yet reacted.)

LEMMA 5.16. *Every positive-continuous piecewise rational linear function $f : \mathbb{R}_{\geq 0}^k \rightarrow \mathbb{R}_{\geq 0}$ is stably computable by a feedforward CRC.*

PROOF. The CRC will have input species X_1, \dots, X_k and output species Y^+ . (While it will be helpful to think of a Y^+ and Y^- species, and during the computation the output will be encoded in their difference, the output of the CRC is only the Y^+ species as per direct computability.)

Let $f : \mathbb{R}_{\geq 0}^k \rightarrow \mathbb{R}_{\geq 0}$ be a positive-continuous piecewise linear function. Since it is positive-continuous, there exist 2^k domains

$$D_U = \{ \mathbf{x} \in \mathbb{R}_{\geq 0}^k \mid (\forall i \in \{1, \dots, k\}) \mathbf{x}(i) > 0 \iff i \in U \},$$

one for each subset $U \subseteq \{1, \dots, k\}$, such that $f \upharpoonright D_U$ is continuous. Define $f_U = f \upharpoonright D_U$. Since D_U is convex, by Lemma 5.15 there is a CRC C_U computing a dual-rail representation $\hat{f}_U : \mathbb{R}_{\geq 0}^k \times \mathbb{R}_{\geq 0}^k \rightarrow \mathbb{R} \times \mathbb{R}$ of f_U . By letting the initial concentration of the “minus” version of the i th input species X_i^- be 0, we convert C_U into a CRC that directly computes an output dual-rail representation of f_U .

The intuition of the proof is as follows: The case $U = \emptyset$ is trivial, as we will have no reactions of the form $\emptyset \rightarrow A$ for any species A , so if no species are initially present, then no species (including Y^+) will ever be produced; this is correct, since any linear function f obeys $f(\mathbf{0}) = 0$. For each non-empty U , we compute f_U independently in parallel by CRC C_U , modifying each reaction producing Y^+ to produce an equivalent amount of species Y_U , which is specific to U . For each such U there are inactive and active “indicator” species J_U and I_U . In parallel, there are reactions that will activate indicator species I_U (i.e., convert J_U to I_U) if and only if all species X_i are present initially for each $i \in U$. These I_U species will then counteract the effect of any CRC computing $f_{U'}$ for $U' \subsetneq U$ by catalytically converting all $Y_{U'}^+$ to Y^- and all $Y_{U'}^-$ to Y^+ . If U is the complete set of indices of non-zero inputs, then only CRCs computing $f_{U'}$ for subsets $U' \subsetneq U$ have produced any amount of Y^+ , so eventually all of these will be counteracted by I_U .

Formally, construct the CRC as follows. Let $l = 2^{k-1} - 1$. For each $i \in \{1, \dots, k\}$, add the reaction $X_i \rightarrow I_{\{i\}} + J_{U_1} + X_i^{U_1} + J_{U_2} + X_i^{U_2} + \dots + J_{U_l} + X_i^{U_l}$, where U_1, U_2, \dots, U_l are all subsets of $\{1, \dots, k\}$ that are strict supersets of $\{i\}$. The extra versions $X_i^{U_1}, \dots, X_i^{U_l}$ of X_i are used as inputs to the parallel computation of each f_U . We generate the inactive indicator species from the input species in this manner, because the CRC is not allowed to start with anything other than the input.

The indicator species are activated as follows: For each nonempty $U, U' \subseteq \{1, \dots, k\}$ such that $U \neq U'$, add the reaction $I_U + I_{U'} + J_{U \cup U'} \rightarrow I_U + I_{U'} + I_{U \cup U'}$.

For each nonempty $U \subseteq \{1, \dots, k\}$, let C_U be the CRC computing an output dual-rail representation of f_U (i.e., dual rail on the output). Modify C_U as follows: Rename the output species of C_U to Y^+ and Y^- , i.e., all parallel CRCs share the same output species. For each reaction producing the output species Y^+ , add the product Y_U^+ (which is a species-specific to C_U) with the same net stoichiometry. Similarly, for each reaction producing the output species Y^- , add the product Y_U^- with the same net stoichiometry. For instance, replace the reaction $A + B \rightarrow Y^+$ by the reaction $A + B \rightarrow Y^+ + Y_U^+$, and replace the reaction $A + Y^+ \rightarrow B + 4Y^+$ by the reaction $A + Y^+ \rightarrow B + 4Y^+ + 3Y_U^+$. Therefore, the eventual amount of Y_U^+ is equal to the total amount of Y^+ produced by C_U , and similarly for Y_U^- and Y^- . For each $U' \subset U$, add the reactions $I_U + Y_{U'}^+ \rightarrow I_U + Y^-$, $I_U + Y_{U'}^- \rightarrow I_U + Y^+$.

Also, for each reaction in C_U , add I_U as a catalyst. This ensures that C_U cannot execute any reactions (and therefore cannot produce any amount of Y^+ or Y^-) unless all species X_i for $i \in U$ are present.

We observe that the dual-rail CRC described above (with output species Y^+ and Y^-) is output-oblivious, as it involves the output species only as products of reactions. Further, the output value is non-negative for any input. Thus, we can convert the dual-rail representation to the direct one (Definition 3.2) with a single output species Y^+ by adding the reaction $Y^+ + Y^- \rightarrow \emptyset$.

To complete the proof, since the CRC of Lemma 5.15 is feedforward, we can confirm by inspection our modifications preserve the feedforward property as well. In particular, one should order species within the CRN dual-computing f_U before any species for supersets of U and after any species for subsets of U . \square

5.5 Negative Result: Directly Computable Functions Are Positive-continuous Piecewise Rational Linear

5.5.1 Siphons and Output Stability. To characterize stable function computation for CRCs, we will crucially rely on the notion of siphons, which we recall from Section 2.4, Definition 2.18. Lemma 5.18 shows the underlying relationship between output stability and siphons.

Let $C = (\Lambda, R, \Sigma, \{Y\})$ be a CRC. We call a siphon Ω *stabilizing* if, for any state \mathbf{d} , $\mathbf{d} \upharpoonright \Omega = \mathbf{0}$ implies that \mathbf{d} is output-stable. In other words, “draining” Ω (removing all of its species) causes the output to stabilize.

LEMMA 5.17. *If \mathbf{c} is an output-stable state, then $\Omega_{\mathbf{c}} = \Lambda \setminus P(\mathbf{c})$ is a stabilizing siphon.*

PROOF. By Lemma 2.21, $\Omega_{\mathbf{c}}$ is a siphon for any state \mathbf{c} . We show the contrapositive that if $\Omega_{\mathbf{c}}$ is not stabilizing, then \mathbf{c} is not output-stable. Suppose there is some particular state \mathbf{d} with $\mathbf{d} \upharpoonright \Omega_{\mathbf{c}} = \mathbf{0}$ that is not output-stable, i.e., for some \mathbf{e} such that $\mathbf{d} \rightsquigarrow \mathbf{e}$, we have $\mathbf{d}(Y) \neq \mathbf{e}(Y)$. Because of this, there must be some reaction α applicable at a state \mathbf{d}' reachable from \mathbf{d} such that α changes the amount of Y (in other words, $\mathbf{M}_{Y,\alpha} \neq 0$). Since $\Omega_{\mathbf{c}}$ is a siphon absent in \mathbf{d} , by Lemma 2.20, $\mathbf{d}' \upharpoonright \Omega_{\mathbf{c}} = \mathbf{0}$, so all reactants of α must be contained in $P(\mathbf{c})$. By Lemma 2.10, we can find a state \mathbf{c}' such that $\mathbf{c} \rightsquigarrow \mathbf{c}'$ and $[\mathbf{c}'] = P(\mathbf{c})$. Since α is applicable at \mathbf{c}' , we see that \mathbf{c} is not output-stable. \square

The next lemma shows the key property of siphons that we will use to reason about stably computing CRC's: that they *characterize* the output-stable states, i.e., the *only* way for the output to stabilize is to drain some stabilizing siphon.

LEMMA 5.18. *There is a set of stabilizing siphons \mathcal{S} such that a state \mathbf{c} is output-stable if and only if $\exists \Omega \in \mathcal{S}$ such that $\mathbf{c} \upharpoonright \Omega = \mathbf{0}$.*

PROOF. Take \mathcal{S} to be $\{\Omega_{\mathbf{c}} \subseteq \Lambda \mid \mathbf{c} \text{ is output-stable}\}$, where $\Omega_{\mathbf{c}} = \Lambda \setminus P(\mathbf{c})$. By Lemma 5.17, each $\Omega \in \mathcal{S}$ is a stabilizing siphon, so if $\exists \Omega \in \mathcal{S}$ with $\mathbf{d} \upharpoonright \Omega = \mathbf{0}$, then \mathbf{d} is output-stable. However, if \mathbf{d} is some output-stable state, then by the definition of $\Omega_{\mathbf{d}}$ (the set of species that cannot be produced from \mathbf{d}), we have $\mathbf{d} \upharpoonright \Omega_{\mathbf{d}} = \mathbf{0}$. By our construction of \mathcal{S} , we know that $\Omega_{\mathbf{d}} \in \mathcal{S}$. \square

5.5.2 Linearity Restricted to Inputs Draining a Siphon. This section aims to prove that the function computed by C , when restricted to inputs that can drain a particular stabilizing siphon, is linear. This is the first step establishing the “piecewise linear” portion of the “positive-continuous piecewise rational linear” claims in Theorem 5.9.

Recall that Ψ , defined in Definition 2.4, is the space of all prepaths (i.e., the vector space in which paths live), and Γ_{∞} , defined in Definition 2.5, is the space of all paths (allowing infinitely many line segments). We now define a map \mathbf{o} that intuitively sends a path to the final state that it reaches.

LEMMA 5.19. *The map $\mathbf{o} : \Psi \rightarrow \mathbb{R}^\Lambda$ sending*

$$\gamma \mapsto \lim_{n \rightarrow \infty} \mathbf{x}_n(\gamma)$$

is linear (recall $\mathbf{x}_n(\gamma)$, defined in Definition 2.4, is the state reached after traversing n segments along γ).

PROOF. To check that \mathbf{o} is a linear function, note that for any $\gamma_0, \gamma_1 \in \Psi$ and $\lambda \in \mathbb{R}$, we have

$$\begin{aligned} \mathbf{o}(\gamma_0 + \lambda \gamma_1) &= \lim_{n \rightarrow \infty} \mathbf{x}_n(\gamma_0 + \lambda \gamma_1) \\ &= \lim_{n \rightarrow \infty} \mathbf{x}_n(\gamma_0) + \lambda \mathbf{x}_n(\gamma_1) \\ &= \lim_{n \rightarrow \infty} \mathbf{x}_n(\gamma_0) + \lambda \lim_{n \rightarrow \infty} \mathbf{x}_n(\gamma_1) \\ &= \mathbf{o}(\gamma_0) + \lambda \mathbf{o}(\gamma_1). \end{aligned}$$

□

Definition 5.20. Let Ω be a stabilizing siphon. Define $\Gamma(\Omega)$ to consist of paths $\gamma \in \Gamma_\infty$ such that $\mathbf{o}(\gamma) \upharpoonright \Omega = \mathbf{0}$.

These are the paths that converge to a state where a given stabilizing siphon is drained.

LEMMA 5.21. $\Gamma(\Omega)$ is convex for each stabilizing siphon Ω .

PROOF. Suppose that γ_0 and γ_1 are in $\Gamma(\Omega)$ and let $\gamma_\lambda = (1 - \lambda)\gamma_0 + \lambda\gamma_1$. By Lemma 2.7, we know that γ_λ is in Γ_∞ . Moreover,

$$\mathbf{o}(\gamma_\lambda) = (1 - \lambda)\mathbf{o}(\gamma_0) + \lambda\mathbf{o}(\gamma_1).$$

Since Ω is drained at both $\mathbf{o}(\gamma_0)$ and $\mathbf{o}(\gamma_1)$, we conclude that it must also be drained at $\mathbf{o}(\gamma_\lambda)$, so $\gamma_\lambda \in \Gamma(\Omega)$. □

Definition 5.22. Let

$$\Sigma(\Omega) = \{\mathbf{x} \in \mathbb{R}_{\geq 0}^\Lambda \mid [\mathbf{x}] \subseteq \Sigma, (\exists \mathbf{o}) \mathbf{x} \rightsquigarrow \mathbf{o}, \text{ and } \mathbf{o} \upharpoonright \Omega = \mathbf{0}\}$$

denote those input states from which the siphon Ω is drainable.

LEMMA 5.23. *Let $f : \mathbb{R}_{\geq 0}^k \rightarrow \mathbb{R}_{\geq 0}$ be stably computed by a CRC $C = (\Lambda, R, \Sigma, \{Y\})$. Let Ω be a stabilizing siphon. Then f restricted to $\Sigma(\Omega)$ is a linear function.*

PROOF. Recall $\Gamma(\Omega)$ from Definition 5.20, the map \mathbf{x}_0 from Definition 2.4, and the map \mathbf{o} from Lemma 5.19. First project $\Gamma(\Omega)$ to $\mathbb{R}_{\geq 0}^\Lambda \times \mathbb{R}_{\geq 0}^\Lambda$ by the map $\gamma \mapsto (\mathbf{x}_0(\gamma), \mathbf{o}(\gamma))$. Let $G \subseteq \mathbb{R}^{k+1}$ be the further projection to the $(k + 1)$ -dimensional subspace corresponding only to the input species X_1, \dots, X_k and output species Y . G is the graph of the function $y = f(\mathbf{x})$ restricted to inputs $\mathbf{x} \in \Sigma(\Omega)$. Since G is the image of a convex set under a linear transformation, it is also convex. We claim that G must be a subset of a k -dimensional hyperplane.

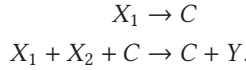
For the sake of contradiction, suppose not. Then there are $k + 1$ non-coplanar points in G . Since G is convex, it contains the entire $(k + 1)$ -dimensional convex hull H of these points. Since H is a $(k + 1)$ -dimensional convex polytope, it contains two different values of y corresponding to the same value of \mathbf{x} , contradicting the fact that only a single y value exists in all output-stable states reachable from \mathbf{x} . This establishes the claim that G must be a subset of a k -dimensional hyperplane.

Since the graph of f is a subset of a k -dimensional hyperplane, f is an affine function. Since there are no reactions of the form $\emptyset \rightarrow \dots, Y$ cannot be produced from the initial state $\mathbf{x} = \mathbf{0}$ (nor can any other species), so $f(\mathbf{0}) = 0$. Therefore, this hyperplane passes through the origin, so it defines a linear function. □

In Lemma 5.23, the reason that we restrict attention to a single output siphon Ω is that if different output siphons are drained, then different linear functions may be computed by the CRC. For example, $X_1 + X_2 \rightarrow Y$ computes $f(x_1, x_2) = x_1$ if siphon $\{X_2\}$ is drained and $f(x_1, x_2) = x_2$ if siphon $\{X_1\}$ is drained. If a CRC stably computes, then an output-stable state is reachable from any input state. Thus, by Lemma 5.18, from every input state some stabilizing siphon is drainable, and the following corollary is immediate:

COROLLARY 5.24. *Let $f : \mathbb{R}_{\geq 0}^k \rightarrow \mathbb{R}_{\geq 0}$ be stably computed by a CRC. Then f is piecewise linear.*

5.5.3 Positive-continuity. Ideally, to prove that the function stably computed by a CRC is positive-continuous, we would like to prove the following: For any stabilizing siphon Ω , the set $\Sigma(\Omega)$ of input states that can drain Ω is closed relative to the positive orthant. If that were true, then we could use a fundamental topological result that if a function is piecewise continuous with finitely many pieces (e.g., piecewise linear), and if the domain defining each piece is closed (with agreement between pieces on intersecting domains), then the whole function is continuous. However, the above statement is not true in general. Consider the following counterexample:



If initially $i(X_1) > i(X_2)$, then the stabilizing siphon $\{X_2\}$ is drainable, by producing $(i(X_1) - i(X_2))/2$ of C via the first reaction (leaving an excess of X_1 over X_2 still), then running the second reaction until X_2 is gone to produce Y . Because X_2 can only be consumed if C is produced, which requires consuming a positive amount of X_1 , the set of inputs from which $\{X_2\}$ can be drained is the non-closed set $\{i \mid i(X_1) > i(X_2)\}$. Note that the above CRC does not stably compute anything because, starting from a state with $i(X_1) > i(X_2)$, the first reaction could run until X_2 exceeds X_1 before starting the second reaction, which would imply that the amount of Y produced depends on how much reaction 1 happens. It is still unclear whether such counterexamples exist for CRCs stably computing some function that is not identically 0.

Instead of relying on $\Sigma(\Omega)$ being closed, we must make a more careful argument. In lieu of working directly with the sets $\Sigma(\Omega)$, we consider “shifted” sets $\tilde{\Sigma}_{(y,z)}(\Omega)$ (see Definition 5.28 below). Each $\tilde{\Sigma}_{(y,z)}(\Omega)$ is (possibly strictly) contained in the original $\Sigma(\Omega)$, but they still cover the set of inputs. Crucially, we are able to show that the shifted sets $\tilde{\Sigma}_{(y,z)}(\Omega)$ are closed, allowing us to apply the argument at the start of this section to prove that every function stably computed by a CRC is positive-continuous.

Definition 5.25. Let

$$X(\Omega) = \left\{ \mathbf{x} \in \mathbb{R}_{\geq 0}^\Lambda \mid (\exists \mathbf{o}) \mathbf{x} \rightarrow^1 \mathbf{o} \text{ and } \mathbf{o} \upharpoonright \Omega = \mathbf{0} \right\}$$

denote those states from which siphon Ω is drainable via a single straight line segment.

LEMMA 5.26. *Let Ω be a siphon. Let $\mathbf{a}_1, \mathbf{a}_2, \dots \in X(\Omega)$ be a convergent sequence of states, where $\mathbf{a} = \lim_{i \rightarrow \infty} \mathbf{a}_i$. Suppose $[\mathbf{a}] = \Lambda$. Then $\mathbf{a} \in X(\Omega)$.*

PROOF. Consider the set P of $\boldsymbol{\gamma} = (\mathbf{x}_0, \mathbf{u}) \in \mathbb{R}_{\geq 0}^\Lambda \times \mathbb{R}_{\geq 0}^R$ such that $\mathbf{o}(\boldsymbol{\gamma}) \in \mathbb{R}_{\geq 0}^\Lambda$ and $\mathbf{o}(\boldsymbol{\gamma}) \upharpoonright \Omega = \mathbf{0}$. Note that reactions occurring with positive flux in \mathbf{u} might not be applicable at \mathbf{x}_0 . P is cut out by a system of non-strict linear inequalities (in other words, it is a polyhedron). By Reference [53], $\mathbf{x}_0(P)$ is also a polyhedron, and is in particular closed.

Note that $X(\Omega) \subseteq \mathbf{x}_0(P)$ and $\mathbf{x}_0(P) \cap \mathbb{R}_{> 0}^\Lambda = X(\Omega) \cap \mathbb{R}_{> 0}^\Lambda$. The first relation follows, since if $\mathbf{x} \in X(\Omega)$, then the straight-line path draining Ω produces a $\boldsymbol{\gamma} \in P$ such that $\mathbf{x}_0(\boldsymbol{\gamma}) = \mathbf{x}$. The

second relation holds, since if every species is present in \mathbf{x}_0 then every reaction is applicable at \mathbf{x}_0 , so any of the points $\gamma \in P$ that project to \mathbf{x}_0 are valid paths that drain Ω .

Since $\mathbf{a} = \lim_{i \rightarrow \infty} \mathbf{a}_i$, we have that for all but finitely many i , $\mathbf{a}_i \in X(\Omega) \cap \mathbb{R}_{>0}^\Lambda$. As a result, these \mathbf{a}_i are in $\mathbf{x}_0(P)$, so \mathbf{a} is in $\mathbf{x}_0(P)$, too, since the set is closed. Since $\mathbf{a} \in \mathbf{x}_0(P) \cap \mathbb{R}_{>0}^\Lambda$, we conclude that $\mathbf{a} \in X(\Omega)$. \square

Note that the hypothesis $[\mathbf{a}] = \Lambda$ is necessary. Otherwise, consider the reactions $A \rightarrow C$, $A+B \rightarrow \emptyset$, and $F+C \rightarrow C$, with $\mathbf{a}_i(C) = 0$, $\mathbf{a}_i(F) = 1$, $\mathbf{a}_i(B) = 1$, and $\mathbf{a}_i(A)$ approaching 1 from above as $i \rightarrow \infty$ (whence $C \notin [\mathbf{a}]$). Then the siphon $\Omega = \{A, B, F\}$ is drainable from each \mathbf{a}_i by running $A \rightarrow C$ until A and B have the same concentration, then running the other two reactions to completion. However, $\mathbf{a}(A) = \mathbf{a}(B)$, so running any amount of reaction $A \rightarrow C$ prevents reaction $A+B \rightarrow \emptyset$ from draining B . Therefore, $\mathbf{a} \notin X(\Omega)$ but $\mathbf{a}_i \in X(\Omega)$ for all i .

Definition 5.27. A pair $(\mathbf{y}, \mathbf{z}) \in \mathbb{R}_{\geq 0}^\Lambda \times \mathbb{R}_{\geq 0}^\Lambda$ of states is a *full input pair* if $[\mathbf{y}] = \Sigma$, $[\mathbf{z}] = \Lambda$, and $\mathbf{y} \rightsquigarrow \mathbf{z}$.

Definition 5.28. If (\mathbf{y}, \mathbf{z}) is a full input pair and Ω is a stabilizing siphon, then define

$$\tilde{\Sigma}_{(\mathbf{y}, \mathbf{z})}(\Omega) = \left\{ \mathbf{x} \in \mathbb{R}_{\geq 0}^\Lambda \mid [\mathbf{x}] = \Sigma \text{ and } \forall \lambda > 0 \text{ such that } \lambda \mathbf{y} < \mathbf{x}, \right. \\ \left. \text{it is the case that } \mathbf{x} - \lambda \mathbf{y} + \lambda \mathbf{z} \in X(\Omega) \right\}.$$

Intuitively, in the CRC at the beginning of the section, the obstacle to the set $\Sigma(\Omega)$ being closed for the siphon $\Omega = \{X_2\}$ was that not all species were present initially: To drain Ω , you first need to produce some (arbitrarily small) amount of C , which leads to the requirement $i(X_1) > i(X_2)$. To fix this problem, $\tilde{\Sigma}_{(\mathbf{y}, \mathbf{z})}(\Omega)$ considers only the states that can drain Ω after having been “perturbed” into a state where all species are present, where the full input pair (\mathbf{y}, \mathbf{z}) specifies how to perform this perturbation.

To see how this works in the example CRC, let (\mathbf{y}, \mathbf{z}) be the full input pair where $\mathbf{y} = 1X_1, 1X_2, 0C, 0Y$ and $\mathbf{z} = \{.25X_1, .75X_2, .5C, .25Y\}$. Then for an input state $\mathbf{x} = \{aX_1, bX_2, 0C, 0Y\}$, $\lambda \mathbf{y} < \mathbf{x}$ when $\lambda < \min(a, b)$. As a result, $\mathbf{x} \in \tilde{\Sigma}_{(\mathbf{y}, \mathbf{z})}(\Omega)$ if $\mathbf{x} - \lambda \mathbf{y} + \lambda \mathbf{z} = (a - .75\lambda)X_1, (b - .25\lambda)X_2, .5\lambda C, .25\lambda Y$ can drain X_2 for all $0 < \lambda < \min(a, b)$. This happens when $b - .25\lambda \leq a - .75\lambda$, so $b + .5\lambda \leq a$. Taking the limit as $\lambda \rightarrow 0$, we see that $b \leq a$, so $\min(a, b) = b$, and taking the limit as $\lambda \rightarrow \min(a, b) = b$, we see that $1.5b \leq a$. Since $1.5b \leq a$ is also a sufficient condition for \mathbf{x} to be in $\tilde{\Sigma}_{(\mathbf{y}, \mathbf{z})}(\Omega)$, we see that $\tilde{\Sigma}_{(\mathbf{y}, \mathbf{z})}(\Omega)$ is closed and contained in $\Sigma(\Omega)$. The next two lemmas show that these properties hold in general.

LEMMA 5.29. *For any full input pair (\mathbf{y}, \mathbf{z}) and any stabilizing siphon Ω , $\tilde{\Sigma}_{(\mathbf{y}, \mathbf{z})}(\Omega)$ is closed relative to $\mathbb{R}_{>0}^\Sigma$.*

PROOF. Let \mathbf{x} be a state such that $[\mathbf{x}] = \Sigma$ and let $\{\mathbf{x}_i\}$ be a sequence such that $\mathbf{x} = \lim_{i \rightarrow \infty} \mathbf{x}_i$ and $\mathbf{x}_i \in \tilde{\Sigma}_{(\mathbf{y}, \mathbf{z})}(\Omega)$. For any $\lambda > 0$ such that $\lambda \mathbf{y} < \mathbf{x}$, by throwing out finitely many terms in the sequence $\{\mathbf{x}_i\}$, we can guarantee that $\lambda \mathbf{y} < \mathbf{x}_i$ for all i , too. Since $\mathbf{x}_i \in \tilde{\Sigma}_{(\mathbf{y}, \mathbf{z})}(\Omega)$, we know that $\mathbf{x}_i - \lambda \mathbf{y} + \lambda \mathbf{z} \in X(\Omega)$ for all i . Since $X(\Omega)$ is closed, we see that

$$\mathbf{x} - \lambda \mathbf{y} + \lambda \mathbf{z} = \left(\lim_{i \rightarrow \infty} \mathbf{x}_i \right) - \lambda \mathbf{y} + \lambda \mathbf{z} = \lim_{i \rightarrow \infty} (\mathbf{x}_i - \lambda \mathbf{y} + \lambda \mathbf{z})$$

is also in $X(\Omega)$. Since this is true for every λ such that $\lambda \mathbf{y} < \mathbf{x}$, we conclude that $\mathbf{x} \in \tilde{\Sigma}_{\mathbf{y}, \mathbf{z}}(\Omega)$. Since this is true for any \mathbf{x} in $\mathbb{R}_{>0}^\Sigma$, we conclude that $\tilde{\Sigma}_{\mathbf{y}, \mathbf{z}}(\Omega)$ is closed relative to $\mathbb{R}_{>0}^\Sigma$. \square

The following lemma is almost immediate from the definition. The only possible concern one might have is that an input state \mathbf{x} is contained in $\tilde{\Sigma}_{(\mathbf{y}, \mathbf{z})}(\Omega)$ “vacuously”—in other words, that there simply does not exist a $\lambda > 0$ such that $\lambda \mathbf{y} < \mathbf{x}$.

LEMMA 5.30. *For any full input pair (y, z) and any stabilizing siphon Ω , $\tilde{\Sigma}_{(y,z)}(\Omega) \subseteq \Sigma(\Omega)$.*

PROOF. Let x be in $\tilde{\Sigma}_{(y,z)}$. By definition, $[y] = \Sigma$, so the following is a well-defined real number:

$$\lambda_0 = \min_{S \in \Sigma} \left\{ \frac{x(S)}{y(S)} \right\}.$$

In other words, λ_0 is the number so $\lambda y < x$ if and only if $\lambda < \lambda_0$. Because $[x] = \Sigma$, we know that $\lambda_0 > 0$. Also $(\lambda_0/2)y \rightsquigarrow (\lambda_0/2)z$ because (y, z) is a full input pair. Then, by additivity of \rightsquigarrow , $x \rightsquigarrow x - (\lambda_0/2)y + (\lambda_0/2)z \rightarrow^1 o$, where Ω is drained at o . Because \rightsquigarrow is transitive (Corollary 2.16), we conclude that $x \rightsquigarrow o$, so $x \in \Sigma(\Omega)$. \square

The above two lemmas show that $\tilde{\Sigma}_{(y,z)}(\Omega)$ is topologically better behaved than $\Sigma(\Omega)$, although possibly smaller. However, for these sets to be useful for analyzing the behavior of a CRC, we need to show that they are not “too small”—in particular, the next lemma shows that under the assumption that a CRC stably computes a function then for any fixed choice of a full input pair (y, z) , the sets $\tilde{\Sigma}_{(y,z)}(\Omega)$ are still big enough to cover all of $\mathbb{R}_{>0}^\Sigma$.

LEMMA 5.31. *If C is a stably computing CRC, then for any fixed full input pair (y, z) , as Ω varies among all of the stabilizing siphons, the sets $\tilde{\Sigma}_{(y,z)}(\Omega)$ cover $\mathbb{R}_{>0}^\Sigma$.*

PROOF. Let x be an input state such that $[x] = \Sigma$ and let λ_0 be as in the proof of Lemma 5.30. We know that $\lambda_0 y \rightsquigarrow \lambda_0 z$ because (y, z) is a full input pair. By additivity of \rightsquigarrow ,

$$x = (x - \lambda_0 y) + \lambda_0 y \rightsquigarrow (x - \lambda_0 y) + \lambda_0 z$$

so, since C is a stably-computing CRC there must be some output-stable state o such that $x - \lambda_0 y + \lambda_0 z \rightsquigarrow o$. By Lemma 5.18, there is an stabilizing siphon Ω so Ω is drained at o .

For any $\lambda > 0$ such that $\lambda y < x$, we know that $\lambda < \lambda_0$, so

$$\begin{aligned} x - \lambda y + \lambda z &= (x - \lambda_0 y + \lambda z) + (\lambda_0 - \lambda)y \\ &\rightsquigarrow (x - \lambda_0 y + \lambda z) + (\lambda_0 - \lambda)z = x - \lambda_0 y + \lambda_0 z. \end{aligned}$$

Since $x - \lambda_0 y + \lambda_0 z \rightsquigarrow o$, we see that $x - \lambda y + \lambda z \rightsquigarrow o$, and, since $[z] = \Lambda$, by Lemma 2.13, we conclude that $x - \lambda y + \lambda z \rightarrow^1 o$. Since this is true for any $\lambda > 0$ such that $\lambda y < x$, we conclude that $x \in \tilde{\Sigma}_{(y,z)}(\Omega)$. This shows that every x with $[x] = \Sigma$ is in $\tilde{\Sigma}_{(y,z)}(\Omega)$ for some Ω , as desired. \square

We now use the above technical machinery to prove the following result, which is *almost* the full negative result for direct computation, but leaves out the constraint that f is *rational* linear. Rationality is shown in Section 5.5.4 below. Recall the *positive-continuous* functions from Definition 5.8.

LEMMA 5.32. *Let $f : \mathbb{R}_{\geq 0}^k \rightarrow \mathbb{R}_{\geq 0}$ be stably computed by a CRC. Then f is positive-continuous and piecewise linear.*

PROOF. Piecewise linearity follows from Corollary 5.24. For positive continuity, we proceed as follows: Let $U \subseteq \{1, \dots, k\}$, let $x \in D_U$ (where D_U is as defined in Definition 5.8), and let $x_1, x_2, \dots \in D_U$ be an infinite sequence of points such that $\lim_{i \rightarrow \infty} x_i = x$. It suffices to show that $\lim_{i \rightarrow \infty} f(x_i) = f(x)$ —i.e., that f is continuous on D_U . We take x_i and x equivalently to represent an initial state of the CRC giving the concentrations of species in $\Sigma = \{X_1, \dots, X_k\}$.

In analyzing the behavior of the CRC on states in D_U , it will help us to consider the functionally equivalent CRC in which we remove species that are not producible from any state in D_U . For the purposes of this proof, we consider this reduced CRC, and let Λ be the corresponding reduced set of species.

Let (\mathbf{y}, \mathbf{z}) be some full input pair. Then, as Ω varies among the stabilizing siphons, $\tilde{\Sigma}_{(\mathbf{y}, \mathbf{z})}(\Omega)$ gives a finite collection of closed sets covering $\mathbb{R}_{>0}^{\Sigma}$ by Lemmas 5.31 and 5.29. Since $\tilde{\Sigma}_{(\mathbf{y}, \mathbf{z})}(\Omega) \subseteq \Sigma(\Omega)$ by Lemma 5.30 and, since f is linear (and therefore continuous) on $\Sigma(\Omega)$ by Lemma 5.23, we see that f is continuous on each of the closed sets in this covering. By Reference [40], if a topological space is a union of finitely many closed sets and f_i are continuous function on each closed set that agree on overlaps, then they combine to give a continuous function. From this result, we conclude that f is continuous on D_U , as desired. \square

5.5.4 Rationality. Recall Definition 5.6 defining rational linear functions and Definition 5.7 defining piecewise rational linear functions.

The main ideas of this section are as follows: To show that a function is piecewise rational linear, we need to show that it is rational linear on some finite set of domains that cover the input space.

A linear function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ that sends \mathbb{Q}^n to \mathbb{Q} is necessarily rational linear. Since a linear function on \mathbb{R}^n is completely determined by its behavior on any open ball, we can check this condition “locally” on any domain that contains an open ball. (Since f is continuous and all of the points of domains that do not contain an open ball are limit points of the other domains, we can ignore domains that do not contain open balls.) The fact that the function sends \mathbb{Q}^n to \mathbb{Q} on such a domain is ultimately a consequence of the fact that the stoichiometry matrix of a CRN has only integer coefficients, so it preserves rationality.

Recall that Ψ , defined in Definition 2.4, is the space of all prepaths, and Γ_{∞} , defined in Definition 2.5, is the space of all paths.

Definition 5.33. A path $\gamma \in \Gamma_{\infty}$ is a *rational path* if it has rational initial concentrations and all of its segments have rational fluxes. In other words, $\mathbf{x}_0(\gamma) \in \mathbb{Q}^{\Lambda}$ and $\mathbf{u}_i(\gamma) \in \mathbb{Q}^R$ for all i .

Note that, since the stoichiometry matrix is an integer-valued matrix, it is automatically the case that $\mathbf{o}(\gamma)$ and every $\mathbf{x}_i(\gamma)$ is in \mathbb{Q}^{Λ} for any rational path γ .

Definition 5.34. We say that two prepaths $\gamma, \gamma' \in \Psi$ have the same sign if for all species S , reactions α , and $i \in \mathbb{N}$, it is the case that $\text{sgn } \mathbf{x}_i(\gamma)_S = \text{sgn } \mathbf{x}_i(\gamma')_S$ and $\text{sgn } \mathbf{u}_i(\gamma)_{\alpha} = \text{sgn } \mathbf{u}_i(\gamma')_{\alpha}$.

LEMMA 5.35. Let $\gamma \in \Gamma_{\infty}$ be a finite piecewise linear path. Then for any $\varepsilon > 0$, there is a rational path $\gamma' \in \Gamma_{\infty}$ such that γ' has the same sign as γ and $\|\gamma' - \gamma\| < \varepsilon$. If γ already has rational initial concentrations, then γ' can be chosen with the same initial concentrations.

PROOF. First, let N be the largest natural number such that $\mathbf{u}_N(\gamma) \neq 0$. (Such an N exists, since γ is finite.) For any reaction $\alpha \in R$ and any $n \in \mathbb{N}_{>0}$ such that $\mathbf{u}_n(\gamma)_{\alpha} = 0$, set $\mathbf{u}_n(\gamma)_{\alpha} = 0$. Now for each species $S \in \Lambda$ such that $\mathbf{x}_k(\gamma)_S = 0$ and $k \leq N$, consider the following linear equation:

$$\mathbf{x}_0(\gamma')_S + \sum_{\substack{1 \leq n \leq k \\ \alpha \in R \\ \mathbf{u}_n(\gamma)_{\alpha} \neq 0}} M_{S\alpha} \mathbf{u}_n(\gamma')_{\alpha} = 0.$$

Aggregating these equations for all $0 \leq k \leq N$ and all $S \in \Lambda$ such that $\mathbf{x}_k(\gamma)_S = 0$ gives a system of equations, linear in $\mathbf{x}_0(\gamma')_S$ and $\mathbf{u}_n(\gamma')_{\alpha}$, with rational coefficients. This equation has a real-valued solution, namely, $\mathbf{x}_0(\gamma') = \mathbf{x}_0(\gamma)$ and $\mathbf{u}_n(\gamma') = \mathbf{u}_n(\gamma)$, so by Lemma C.1 (proven in Appendix C), it must have a solution with rational coefficients that is δ -close for any $\delta > 0$. By taking δ small enough, we can of course make $\delta < \varepsilon$, but we can also guarantee that $\mathbf{x}_0(\gamma')_S$ is positive whenever $\mathbf{x}_0(\gamma)_S$ is positive and similarly for $\mathbf{u}_n(\gamma')_{\alpha}$. We have therefore specified a $\gamma' \in \Psi$ with the same sign as γ . Since γ' has the same sign as γ , and, since γ is a valid path, we conclude that γ' is also a valid path, so $\gamma' \in \Gamma_{\infty}$.

If $\mathbf{x}_0(\gamma)$ is already in \mathbb{Q}^Δ , then the same argument applies, with the modification that you fix $\mathbf{x}_0(\gamma') = \mathbf{x}_0(\gamma)$, and instead solve the inhomogeneous system of equations

$$\sum_{\substack{1 \leq n \leq k \\ \alpha \in R \\ \mathbf{u}_n(\gamma)_\alpha \neq 0}} M_{S\alpha} \mathbf{u}_n(\gamma')_\alpha = -\mathbf{x}_0(\gamma)_S$$

when $\mathbf{x}_k(\gamma)_S = 0$. □

LEMMA 5.36. *Let Ω be a stabilizing siphon. If $\Sigma(\Omega)$ contains an open ball, then f is rational linear when restricted to inputs in $\Sigma(\Omega)$.*

PROOF. Let B be the open ball contained in $\Sigma(\Omega)$ and let \mathbf{x} be in $\mathbb{Q}^\Delta \cap B$. We know that there is a piecewise linear path γ starting at \mathbf{x} such that $f(\mathbf{x}) = \mathbf{o}(\gamma)_Y$. By Theorem 2.15, we may assume without loss of generality that γ is finite. By Lemma 5.35, there is a rational path γ' with the same sign as γ such that $\mathbf{x}_0(\gamma') = \mathbf{x}_0(\gamma) = \mathbf{x}$. Because $\mathbf{o}(\gamma)$ is an output-stable state, some siphon Ω is drained at $\mathbf{o}(\gamma)$. Since γ' has the same sign as γ , we know that Ω is also drained at $\mathbf{o}(\gamma')$, so $\mathbf{o}(\gamma')$ is also output-stable. We must then have that $f(\mathbf{x}) = \mathbf{o}(\gamma')_Y$, but by the construction of γ' , we know that $\mathbf{o}(\gamma')_Y \in \mathbb{Q}$. Since $B \subseteq \Sigma(\Omega)$, we know that $f|_B$ is linear by Lemma 5.23. Since B is an open ball, we know that $\mathbb{Q}^\Delta \cap B$ contains a basis for \mathbb{Q}^Δ , so $f|_B$ is a linear function that maps \mathbb{Q}^Δ to \mathbb{Q} . Since every linear function $\mathbb{R}^\Delta \rightarrow \mathbb{R}$ that sends \mathbb{Q}^Δ to \mathbb{Q} is rational linear, we are done. □

Recall that a *closed domain* is the closure of an open set.

LEMMA 5.37. *Let $f : X \rightarrow Y$ be a continuous function defined piecewise on closed sets, so X is covered by finitely many closed sets $D_1 \dots D_k$ and there are continuous functions $g_i : X \rightarrow Y$ such that $f|_{D_i} = g_i|_{D_i}$. Then there are (possibly empty) closed domains $E_1 \dots E_k$ that cover X such that $f|_{E_i} = g_i|_{E_i}$.*

PROOF. We show how to convert each D_i that is not a closed domain to a corresponding E_i that is. Let D_i be some set that is not a closed domain. Let

$$D_+ = \bigcup_{j \neq i} D_j.$$

Note that $D_1, \dots, D_{i-1}, X \setminus D_+, D_{i+1}, \dots, D_k$ cover X . Let E_i be the closure of $X \setminus D_+$. Clearly, E_i is a closed domain. Since our original sets $D_1 \dots D_k$ cover X , we know that $X \setminus D_+ \subseteq D_i$, and, since D_i is closed, this implies that $E_i \subseteq D_i$. Because of this, we also have that $f|_{E_i} = g_i|_{E_i}$. Finally, because $X \setminus D_+ \subseteq E_i$, we know that the sets $D_1, \dots, D_{i-1}, E_i, D_{i+1}, \dots, D_k$ cover X . □

The following is the main result of Section 5.5, showing a limitation on the computational power of CRCs stably computing functions in the direct sense.

LEMMA 5.38. *Let $f : \mathbb{R}_{\geq 0}^k \rightarrow \mathbb{R}_{\geq 0}$ be stably computed by a CRC $C = (\Lambda, R, \Sigma, \{Y\})$. Then f is positive-continuous and piecewise rational linear.*

PROOF. By Lemma 5.32, we know that f is positive-continuous and piecewise linear. By a general topological argument one could show that any function with these properties has domains of definition that are closed relative to $\mathbb{R}_{> 0}^\Sigma$, but, since by Lemma 5.29 the domains we constructed earlier already have this property, we will not give the general proof here. By Lemma 5.37, we can replace the closed sets that give the domains of definition of f by closed domains. If some of the domains produced by Lemma 5.37 are empty, then we can simply ignore them in what follows. Since all of the nonempty domains are the closures of nonempty open sets, they must each contain

some open ball. By Lemma 5.36, f is a rational linear function when restricted to each of these closed domains, so f is piecewise rational linear on $\mathbb{R}_{>0}^\Sigma$.

For any proper subset U of the input species, one can apply the above argument to the reduced CRN that discards all species not producible from the given inputs to show that f is continuous and piecewise rational linear on D_U . This shows that f is a positive-continuous piecewise rational linear function on all of $\mathbb{R}_{\geq 0}^\Sigma$. \square

5.6 Negative Result: Dual-rail Computable Functions Are Continuous Piecewise Rational Linear

The following result, a dual-rail analog of Lemma 5.36, is not necessary for the proof of the main result of this section (Lemma 5.40), but it may be of independent interest.

PROPOSITION 5.39. *Let $f : \mathbb{R}^k \rightarrow \mathbb{R}$ be stably dual computed by a CRC. Let Ω be a stabilizing siphon. Then f restricted to inputs that have a dual rail representation in $\Sigma(\Omega)$ is linear.*

PROOF. A dual-rail computing CRC can be thought to *directly* compute two separate functions $\hat{f}^+, \hat{f}^- : \mathbb{R}_{\geq 0}^{2k} \rightarrow \mathbb{R}_{\geq 0}$ such that $\hat{f} = \hat{f}^+ - \hat{f}^-$ where \hat{f} is a dual rail representation of f . By Lemma 5.23, we know that \hat{f}^+ and \hat{f}^- are rational linear when restricted to $\Sigma(\Omega)$. The proposition follows, because linearity is closed under subtraction. \square

The following is our main negative result for dual-rail CRC's, a dual-rail analog of Lemma 5.38:

LEMMA 5.40. *Let $f : \mathbb{R}^k \rightarrow \mathbb{R}$ be stably dual-computable by a CRC. Then f is continuous and piecewise rational linear.*

PROOF. Let C be the CRC stably computing a dual-rail representation \hat{f} of f , with input species $X_1^+, \dots, X_k^+, X_1^-, \dots, X_k^-$ and output species Y^+, Y^- .

Similarly to the proof of Proposition 5.39, a dual-rail computing CRC can be thought to *directly* compute two separate functions $\hat{f}^+, \hat{f}^- : \mathbb{R}_{\geq 0}^{2k} \rightarrow \mathbb{R}_{\geq 0}$ such that $\hat{f} = \hat{f}^+ - \hat{f}^-$ where \hat{f} is a dual rail representation of f . Since \hat{f}^+ and \hat{f}^- are stably computed by a CRC, Lemma 5.38 implies that they are both piecewise rational linear, and, since piecewise rational linear functions are closed under subtraction, this implies that f is also piecewise rational linear. It remains to show that f is continuous.

For any input $\mathbf{x}' \in \mathbb{R}^k$ to f , there is an initial state $\mathbf{x} \in \mathbb{R}_{>0}^\Sigma$ representing \mathbf{x}' , with strictly positive concentrations of all input species. (For example, if $\mathbf{x}'(1) = 5$, then we can choose $\mathbf{x}(X_1^+) = 6$ and $\mathbf{x}(X_1^-) = 1$.) Let $\mathbf{x}'_1, \mathbf{x}'_2, \dots \in \mathbb{R}^k$ be any sequence of inputs to f such that $\lim_{i \rightarrow \infty} \mathbf{x}'_i = \mathbf{x}'$. Let us represent each input \mathbf{x}'_i in the sequence by an initial state $\mathbf{x}_i \in \mathbb{R}_{>0}^\Sigma$ such that $\lim_{i \rightarrow \infty} \mathbf{x}_i = \mathbf{x}$. Then the sequence of initial states has the property that all of the \mathbf{x}_i 's obey $[\mathbf{x}_i] = \Sigma$. By Lemma 5.32, f is continuous on the domain in which all input species are positive, which includes the input represented by \mathbf{x} and the inputs represented by all of the \mathbf{x}_i 's. Therefore, $f(\mathbf{x}') = \lim_{i \rightarrow \infty} f(\mathbf{x}'_i)$, so f is continuous. \square

6 EXTENSIONS

6.1 A Game-theoretic Formulation of Rate-independent Computation

In this section, we use our notion of a valid rate schedule (Definition 2.22) to propose a framework for studying rate-independent computation. Intuitively, we consider a model where the CRC experiences intermittent “shocks” where the system behaves erratically and the user of the CRC loses some amount of control of the rates of its reactions. We then say that the CRC rate-independently computes a function if, despite these shocks, the CRC still converges to the correct output.

In principle, one could consider different versions of the above model, depending on how much control the user of the CRC is expected to have over the kinetics of the system and how severe the shocks to the system are expected to be. Our goal in this section is to introduce a framework that is general enough to accommodate these various situations.

We will formalize the situation presented above by describing it as an infinite game, played by two players, which we will call the Demon and the Chemist. Here, the Chemist represents the user of the CRC, who is trying to perform some rate-independent computation, and the Demon represents (per tradition) the forces of nature, human failing, and so on, which create the intermittent shocks to the system. In this game, the Demon and the Chemist take turns building a valid rate schedule. The Chemist wins the game if the amount of the output species converges to the correct value as time goes to infinity. If the Chemist has a winning strategy for the game associated with a given CRC, then we say that the CRC rate-independently computes the desired function.

To model the different levels of control that the Chemist has over the system, and the different levels of severity of the shocks that the Demon can induce in the system, we can consider games where the Demon and the Chemist are only allowed to play moves from some restricted class of valid rate schedules. In this article, we have been interested in functions that are rate-independently computable in a very strong sense, i.e., where the shocks are arbitrarily severe—so the Demon is allowed to play any valid rate schedule (Strong Demon).

Once we have developed the game-theoretic model of rate-independent computation more explicitly, we will be able to use our main result Theorem 5.9 to deduce the following claim: The class of functions that one can compute rate-independently with a Strong Demon is effectively insensitive to the amount of control that the Chemist has over the system. In particular, the class of functions that are rate-independently computable by Strong Demon, Strong Chemist games, where both the Demon and the Chemist can play any valid rate law, is *the same* as any Strong Demon complexity class, where the Chemist is restricted to playing only a subset of valid rate laws.

To formalize the above discussion, we can use the notion of an infinite game [41].

Definition 6.1. An infinite game is a pair of a set X of possible moves and a set $A \subseteq X^{\mathbb{N}}$ called the payoff set.

Intuitively, one should imagine that two players, player I and player II, are taking turns playing moves from the set X . Together they form a sequence $(x_1, x_2, \dots) \in X^{\mathbb{N}}$ and player I wins if this sequence is in A . In most games of interest, the set of moves that a player can make is restricted by the state of the game. The definition above captures this by using a payoff set such that a player will lose instantly if they play a move outside of some “legal” set of moves. For any previously played sequence, we can thus let $\text{moveset } M(x_1, \dots, x_n)$ be the legal set of moves for the two players (depending on whether n is odd or even) and assume that the payoff set A is defined relative to these sets of allowed moves accordingly (i.e., the first time that an illegal move is made, the other party wins). An infinite sequence of moves is consistent with the moveset if every move by both players is legal.

Definition 6.2. For an infinite game (X, A) a winning strategy for player II is a collection of functions $f_{2k} : X^k \rightarrow X$ so the sequence

$$(x_1, f_2(x_1), x_3, f_4(x_1, x_3), x_5, f_6(x_1, x_3, x_5), \dots)$$

is never in A for every choice of x_1, x_3, \dots . A winning strategy for player I is defined similarly.

For our context, fix a CRC C and an initial state $\mathbf{x} \in \mathbb{R}_{\geq 0}^{\Sigma}$. The set of possible moves X is the set of all rate schedules that are zero at times $t > 1$. Such moves (i.e., rate schedules) can be naturally

concatenated into a longer rate schedule: The concatenation $\mathbf{f}_1 \circ \dots \circ \mathbf{f}_n$ is the rate schedule that follows \mathbf{f}_i for time $i-1 \leq t < i$ and zero elsewhere. Our game captures the idea that the Demon wins by making the constructed rate schedule not converge to the desired output, where the Demon is player I and the Chemist is player II. We allow the Demon to play any valid rate schedule, but the Chemist may have varying power as we discuss below.

The most unrestricted moveset contains all possible valid rate schedules. We call this case the Strong Demon Strong Chemist game.

Definition 6.3. For a CRC C and function $f : \mathbb{R}_{\geq 0}^k \rightarrow \mathbb{R}_{\geq 0}$, an initial state $\mathbf{x} \in \mathbb{R}_{\geq 0}^\Sigma$, the *Strong Demon Strong Chemist game* is the game where the moveset $M(\mathbf{f}_1, \dots, \mathbf{f}_n)$ includes all the moves \mathbf{f}_{n+1} such that $\mathbf{f}_1 \circ \dots \circ \mathbf{f}_{n+1}$ is a valid rate schedule starting at \mathbf{x} , and a sequence of moves $(\mathbf{f}_1, \mathbf{f}_2, \dots)$ consistent with the moveset is in the payoff set iff $\lim_{t \rightarrow \infty} \boldsymbol{\rho}(t) \upharpoonright \{Y\} \neq f(\mathbf{x})$ for trajectory $\boldsymbol{\rho}$ corresponding to $\mathbf{f}_1 \circ \mathbf{f}_2 \circ \dots$ starting at \mathbf{x} .

LEMMA 6.4. *The following are equivalent for a CRC C and function $f : \mathbb{R}_{\geq 0}^k \rightarrow \mathbb{R}_{\geq 0}$:*

- (1) *The Chemist has a winning strategy for the Strong Demon Strong Chemist game associated with C and f for every initial state $\mathbf{x} \in \mathbb{R}_{\geq 0}^\Sigma$.*
- (2) *C stably computes f .*

PROOF. First suppose that C stably computes. Then, by Theorem 2.27, we know that for any first move \mathbf{f}_1 the Demon plays, the state $\boldsymbol{\rho}(1)$ will be reachable from the initial state \mathbf{x} . By the definition of stable computation there must be a state \mathbf{z} reachable from \mathbf{x} so \mathbf{z} is output-stable with the correct amount of output. By Lemma 2.26, there is a valid rate schedule \mathbf{f}_2 that goes from \mathbf{x} to \mathbf{z} , and by “rescaling” the rate schedule by increasing the rate of every reaction, we can guarantee that \mathbf{f}_2 is completed within a single unit of time. Since \mathbf{z} is output-stable, every further move $\mathbf{f}_4, \mathbf{f}_6, \dots$ that the Chemist makes can just be taken to be the zero rate schedule (where no reaction occurs). This specifies a winning strategy for the Chemist for the Strong Demon Strong Chemist game associated with C , f , and \mathbf{x} .

However, suppose that C does not stably compute. Then let ϵ , \mathbf{x} , and \mathbf{z} be the ones given by Theorem 3.3. The Demon can play a rate schedule that goes from \mathbf{x} to \mathbf{z} . After this, every time it is the Demon’s turn the CRC will be in a state \mathbf{o} that is reachable from \mathbf{z} , so the Demon will always be able to play a valid rate schedule that takes the CRN to \mathbf{o}' as in Theorem 3.3 with $|\mathbf{o}'(Y) - f(\mathbf{x})| > \epsilon$. This shows that the Demon has a winning strategy for the Strong Demon Strong Chemist game associated with C , f and \mathbf{x} , so certainly the Chemist does not have a winning strategy. \square

We can also consider games where the players are restricted to only playing a subset of the valid rate schedules. We are particularly interested in games where the Demon is allowed to play any valid rate schedule as above, but the Chemist is restricted to only playing rate schedules from a particular restricted moveset M . It turns out that as long as the chemist has *some fair* rate schedule to play, the computational power is the same as with a Strong Chemist.

We first define the validity and fairness restrictions on movesets. Then given a restricted moveset for the Chemist, we define the corresponding game similar to Definition 6.3.

Definition 6.5. A moveset $M(\mathbf{f}_1, \dots, \mathbf{f}_n)$ is valid if it includes only moves \mathbf{f}_{n+1} such that $\mathbf{f}_1 \circ \dots \circ \mathbf{f}_{n+1}$ is a valid rate schedule starting at \mathbf{x} . Further, we say that a valid moveset $M(\mathbf{f}_1, \dots, \mathbf{f}_n)$ is Chemist-fair if there is some $\mathbf{H} : \mathbb{R}_{\geq 0}^\Lambda \rightarrow \mathbb{R}_{\geq 0}^R$ (with the same properties as in Definition 4.1) such that for any odd n , the moveset contains at least one \mathbf{f}_{n+1} such that the trajectory $\boldsymbol{\rho}$ corresponding to $\mathbf{f} = \mathbf{f}_1 \circ \dots \circ \mathbf{f}_{n+1}$ starting at \mathbf{x} satisfies $\mathbf{f}_\alpha(t) \geq \mathbf{H}_\alpha(\boldsymbol{\rho}(t))$ for every reaction α and for all $t \in [n, n+1)$.

Definition 6.6. Consider a CRC C , function $f : \mathbb{R}_{\geq 0}^k \rightarrow \mathbb{R}_{\geq 0}$, and an initial state $\mathbf{x} \in \mathbb{R}_{\geq 0}^\Sigma$. Let $M(\mathbf{f}_1, \dots, \mathbf{f}_n)$ be any valid moveset. We say the *Strong Demon M -Chemist* is the game where a sequence of moves $(\mathbf{f}_1, \mathbf{f}_2, \dots)$ consistent with the moveset is in the payoff set iff $\lim_{t \rightarrow \infty} \boldsymbol{\rho}(t) \upharpoonright \{Y\} \neq f(\mathbf{x})$ for trajectory $\boldsymbol{\rho}$ corresponding to $\mathbf{f}_1 \circ \mathbf{f}_2 \circ \dots$ starting at \mathbf{x} .

It may seem that depending on what moves the Chemist is allowed to play, different classes of functions are computable (i.e., the Chemist has a winning strategy). However, the following theorem shows that the class of functions is invariant to the class of movesets that the Chemist has available, as long as there is always at least one fair move:

Definition 6.7. A *Strong Demon complexity class* SD is a set of functions $f : \mathbb{R}_{\geq 0}^k \rightarrow \mathbb{R}_{\geq 0}$ for all $k \in \mathbb{N}$, that is characterized in the following way: There is a (non-empty) set of Chemist-fair movesets \mathcal{M} so $f : \mathbb{R}_{\geq 0}^k \rightarrow \mathbb{R}_{\geq 0}$ is in SD if and only if there is a CRC C such that for each $M \in \mathcal{M}$ and initial state \mathbf{x} the Chemist has a winning strategy for the associated Strong Demon M -Chemist game.

THEOREM 6.8. *Every Strong Demon complexity class is the same as the Strong Demon Strong Chemist class.*

PROOF. If a Chemist has a winning strategy in a Strong Demon M -Chemist game, then clearly the Chemist has a winning strategy in the Strong Demon Strong Chemist game. Thus, every Strong Demon complexity class is a subset of the Strong Demon Strong Chemist class. For the other direction, let SD be some Strong Demon complexity class, and let \mathcal{M} be the nonempty set of Chemist-fair movesets associated with SD. Suppose that $f : \mathbb{R}_{\geq 0}^k \rightarrow \mathbb{R}_{\geq 0}$ is in the Strong Demon Strong Chemist class. Then, by Lemma 6.4 and Theorem 5.9, we know that f is fairly computable by a CRC C . For any moveset $M \in \mathcal{M}$, let us show that the Chemist has a winning strategy for the Strong Demon M -Chemist game associated with C , and f . Since M is Chemist-fair, we know that there is some function $\mathbf{H} : \mathbb{R}_{\geq 0}^\Lambda \rightarrow \mathbb{R}_{\geq 0}^R$ as in Definition 6.5. On their turn, the Chemist simply plays any move in the moveset M that satisfies the condition on \mathbf{f}_{n+1} in Definition 6.5. This strategy is winning for the Chemist, because the total rate schedule \mathbf{f} obtained by concatenating the \mathbf{f}_i 's is fair, since on the set $T = \bigcup_{n=1}^\infty [2n-1, 2n)$, we know that $\mathbf{f}_\alpha(t) \geq \mathbf{H}_\alpha(\boldsymbol{\rho}(t))$. Since C was chosen to fairly compute f this proves that $\lim_{t \rightarrow \infty} (\boldsymbol{\rho}(t) \upharpoonright \{Y\}) = f(\mathbf{x})$. This shows that $f \in \text{SD}$, so any Strong Demon M -Chemist complexity class is the same as the Strong Demon Strong Chemist complexity class. \square

6.2 Non-zero Initial Context

Throughout this article, we have assumed that the only species allowed to be present at the start of the computation are the input species. Instead, one could consider a model where certain non-input species $Z_1 \dots Z_n$, called the *initial context* [18], have a fixed, nonzero rational concentration at the start of the computation. In this setting, we can clearly compute more functions than in the setting without initial context: For instance, we can easily compute $f(x_1, \dots, x_k) = C$ for some nonzero constant C , which is impossible without initial context because f is affine but not linear.

In fact, we can dual-rail compute any continuous piecewise rational affine function $f : \mathbb{R}^k \rightarrow \mathbb{R}$, i.e., any function that is a rational linear function plus a rational constant: $f(\mathbf{x}) = \mathbf{a} \cdot \mathbf{x} + b$. To see this, first note that we can compute any rational affine function by using the initial context to offset the value at $f(\mathbf{0}) = b$. In fact, we can simply let output species Y^+ and Y^- be the initial context, with $\mathbf{x}(Y^+) = b$ initially if $b > 0$ and $\mathbf{x}(Y^-) = -b$ otherwise. Similar machinery to the proof of Lemma 5.15 can be used to extend this to continuous *piecewise* rational affine functions: By Theorem 5.11, any continuous piecewise rational affine function can be represented in max-min form, and then our construction from Section 5.3 shows that we can compute our given function

f . In the direct computation setting, by a construction like the one in Section 5.4, we can compute any positive-continuous piecewise rational affine function $f : \mathbb{R}_{\geq 0}^k \rightarrow \mathbb{R}_{\geq 0}$.

It also turns out that, even with initial context, we cannot compute any more functions than these. To see this, note that without loss of generality, we can assume that there is only one initial context species Z with initial concentration 1, since we can modify any CRC with initial context to include reactions that convert Z into Z_1, \dots, Z_n with appropriate concentrations.²³ Now let $g(x_1, \dots, x_k, z)$ be the value that the CRC computes when the input species have initial values x_1, \dots, x_k and the initial context species has value z . *A priori*, g is only well-defined when $z = 1$, but because paths remain valid after scaling, we know that

$$g(x_1, \dots, x_k, z) = z \cdot g(x_1/z, \dots, x_k/z, 1) = z \cdot f(x_1/z, \dots, x_k/z)$$

for any value of $z > 0$. This shows that g is well-defined on D_U (recall Definition 5.8) for every U that contains Z , so we can apply the results of Sections 5.5 and 5.6 to characterize g on these domains. Using the fact that $f(x_1, \dots, x_k) = g(x_1, \dots, x_k, 1)$ gives us the desired result.

Since every continuous function on a compact domain is uniformly continuous, it can be uniformly approximated by continuous piecewise rational affine functions. This shows that we can use rate-independent CRNs to approximate continuous functions. Note that for the negative argument above to work, it was important that all of the initial concentrations of the initial context species were rational. For practical purposes, this assumption is not at all restrictive, but it might be of theoretical interest to know what other functions can be computed if the initial concentrations are allowed to be arbitrary real numbers.

7 CONCLUSION AND OPEN QUESTIONS

We characterized the class of functions computable in a manner that is absolutely robust to reaction rates in the continuous model of chemical kinetics. Such rate-independent computation must rely solely on reaction stoichiometry—which reactants, and how many of each, become which products, and how many of each? We considered two methods of encoding inputs and outputs: direct and dual-rail. The dual-rail encoding permits easier composition of modules and can represent negative values; we characterized its computational power as continuous, piecewise rational linear. The direct encoding, however, allows computing functions that are discontinuous at the faces of the nonnegative orthant. For both encodings, we showed matching negative results (showing that nothing more can be computed) and positive results (describing CRNs computing any function in the class).

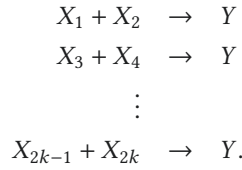
Since rate-independent computation does not require difficult-to-achieve tuning of parameters or reaction conditions, it may be significantly more “engineerable” than rate-dependent computation. More generally, our work also helps uncover the multifaceted sources of chemical computational power by disentangling the control of stoichiometry from reaction rates.

We now describe some natural open questions.

Reaction complexity of stably computable functions. An interesting question regards the description complexity of functions stably computable by CRNs. Some piecewise linear functions have a number of pieces exponential in the number of inputs; for example, $f(x_1, \dots, x_{2k}) = \min(x_1, x_2) + \min(x_3, x_4) + \dots + \min(x_{2k-1}, x_{2k})$ has 2^k linear pieces. If we express this function in $\max \min g_i$ form of Theorem 5.11, then we need 2^k different linear g_i , and thus the construction in the proof of Lemma 5.15 would require exponentially many species and reactions. However, this

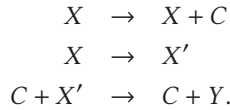
²³For example, to simulate initial context $\{3/2 Z_1, 1/2 Z_2\}$ from $\{1 Z\}$, add the reactions $2Z \rightarrow Z'$ and $Z' \rightarrow 3Z_1 + Z_2$.

particular f has a more succinct CRN that stably computes it, namely, the reactions



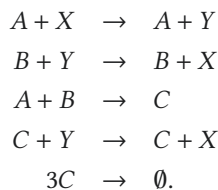
Given a positive-continuous, piecewise rational linear function f , how can we tell whether it has a more compact CRN stably computing it than our construction? If it does, then how can we arrive at it?

Requiring “always” fair rate schedules. Lemma 4.9 shows that for a feedforward CRC that stably computes a function f , any fair rate schedule converges to the correct output of f , where fair essentially means that applicable reactions must have positive rate for an infinite subset of times. In other words, the adversary is allowed for some times outside this subset for the rate schedule to be unfair: to “starve” some applicable reactions by keeping their rates at 0 despite all reactants being present. Since such rate schedules seem physically implausible, it is natural to consider a modified definition of computation, one that requires every rate schedule that is *always-fair* (i.e., for *all* time, applicable reactions must have positive rate) to converge to the correct output. The following CRC shows that these two requirements can result in different behaviors for a given CRC:



With initial concentration x of X , every always-fair rate schedule converges to concentration x of Y , so it computes the identity function $f(x) = x$ under this modified definition. (Note that always-fairness requires the first two reactions to have positive rate at time 0, so C immediately becomes present, at which point it is inevitable to convert all X to X' and all X' to Y .) However, the CRC does not stably compute f : An initially unfair adversary can execute the second reaction to completion, starving the first reaction until it becomes inapplicable. This schedule never produces any C , so Y stays at 0, yet the schedule is fair by the original definition, since, once all reactions become inapplicable, subsequent rate 0 of all reactions for all time vacuously satisfies the definition of fair. It is an interesting question is whether, under the modified definition of always-fair, some CRC can compute a function that is not stably computable.

Arbitrary but fixed rate constants. A related notion of rate-independence is one where the form of rate-law cannot vary, but the constant parameters can, e.g., mass-action rates, where an adversary picks the rate constants (possibly depending on the initial input). For example, consider the following CRN with input species A, B, C, X and output species Y .



Let a, b, c, x denote the initial concentrations of species A, B, C, X . This system does not stably compute any function in the model defined in this article because on input $a = b$, it can stabilize to any value of output y between 0 and x .

In contrast, consider the above system under mass-action kinetics, where an adversary picks the rate constants, but they remain constant over time. Because A and B are required to produce C , and at least one of them goes to 0 by the third reaction, the concentration of C approaches 0 as time goes to infinity by the final reaction, no matter the rate constants. If $a > b$, then also the concentration of B approaches 0 but the concentration of A remains bounded away from 0. Therefore, the output y converges to x , regardless of what the rate constants are. Similarly, if $b > a$, then the output y approaches 0. When $a = b$, the concentrations of A, B, C approach 0 at different rates. The concentrations of A, B are $\Theta(\frac{1}{t})$ at time t (rate of bimolecular decay), and the concentration of C is $\Theta(\frac{1}{\sqrt{t}})$ (rate of trimolecular decay). As a result, the effective rate of conversion of Y to X via the channel $C + Y \rightarrow C + X$ is $\Theta(\frac{1}{\sqrt{t}})$. Since this is $\omega(\frac{1}{t})$ the output y always converges to 0 regardless of the rate constants (in our particular case, the concentration of Y is $e^{-\Theta(\sqrt{t})}$). From the above, this CRN computes $f(a, b, c, x) = x$ when $a > b$ and $f(a, b, c, x) = 0$ when $a \leq b$, no matter what the rate constants are. This function is discontinuous at points where $a = b$, so it is not positive-continuous, thus not stably computable by any CRN under our model of rate-independence. It remains open to classify what functions can be computed by mass-action CRNs in which rate constants are chosen adversarially.

Axiomatic derivation of reachability. An important contribution of this article is to develop segment-reachability as the “correct” notion of rate-independent reachability. While we justify our definition of segment-reachability by its equivalence to valid rate schedules (Theorem 2.27), one could imagine taking an axiomatic approach: Any notion of rate-independent reachability ought to satisfy certain constraints. For instance, it should be transitive. Also, if d is straight-line reachable from c , then it is evidently allowed by stoichiometry for the CRN to evolve from c to d , so d should be reachable from c . Of course, there are many relations that satisfy these two constraints: For instance, the relation where every state is reachable from every other state satisfies these two constraints—but this is evidently too permissive. Indeed, our notion of segment-reachability is the transitive closure of the relation of straight-line reachability (Corollary 2.16), so it is minimal among all of the relations with these two properties. However, since our goal is to develop a notion of reachability that is as unrestricted as possible, a more compelling axiomatic derivation would follow from simple set of natural conditions for which segment-reachability is *maximal*.

Absolute inhibition. In biology it is not uncommon to have rate laws with explicit inhibitors; the higher the concentration of the inhibitor, the slower the reaction. Formally, general rate laws have been described in which reactants are partitioned into consumed species, species that increase the reaction rate (catalysts), and species that decrease the rate (inhibitors) [26]. In our model, while inhibitors can be modeled mechanistically as sequestering reacting species in a non-reactive form (e.g., $A + B \rightarrow C$ is inhibited by I via the reaction $A + I \rightleftharpoons AI$), such inhibition just slows down the reaction rate but does not prevent the reaction entirely. In contrast, one can imagine a definition of reachability in which a reaction is applicable only if all of its reactants are present *and* all of its inhibitors are absent. It remains an open question whether the computational power of rate-independent computation changes if such absolute inhibition is allowed. We note that this change drastically changes the notion of reachability because it is no longer additive: It might be that $x \rightsquigarrow y$, but $x + c \not\rightsquigarrow y + c$ if c contains some inhibitors of reactions occurring in the first path. Further, for discrete CRNs, absolute inhibition dramatically expands computational power of stable computation to Turing universality [21, 38] (i.e., the ability to

compute any function computable by any algorithm). thus it seems reasonable to conjecture that it also expands the computational power in the continuous setting.

Decision problems. In the discrete CRN model, particularly the subset of it known as population protocols [6], a major focus of research is on *decision* problems with a yes/no output, a.k.a. predicates. The typical output convention partitions the set Λ of species into two disjoint subsets $\Lambda = \Lambda_Y \cup \Lambda_N$, the “yes voters” and “no voters.” The goal of stable computation in this setting is to reach a configuration with a unanimous, correct vote (e.g., if the correct answer is yes, then only species in Λ_Y are present), which is also stable in the sense that no incorrect voter is producible. In this setting, it has been shown that exactly the *semilinear* predicates can be stably decided [6, 7] by discrete CRNs.

The concept can be similarly defined with continuous CRNs using our notion of segment-reachability. Say that a CRN with voting species defined as above *stably computes* a predicate $\phi : \mathbb{R}_{\geq 0}^k \rightarrow \{Y, N\}$ if, for any initial configuration $\mathbf{x} \in \mathbb{R}_{\geq 0}^k$, for any configuration \mathbf{c} such that $\mathbf{x} \rightsquigarrow \mathbf{c}$, there is a configuration \mathbf{y} such that $\mathbf{c} \rightsquigarrow \mathbf{y}$, and \mathbf{y} is *stably correct*, meaning that for all \mathbf{y}' such that $\mathbf{y} \rightsquigarrow \mathbf{y}'$, we have $\emptyset \neq [\mathbf{y}'] \subseteq \Lambda_{\phi(\mathbf{x})}$. In other words, stable computation leads to a nonempty configuration with only “correct” votes (according to ϕ), and this is also true of every configuration reachable from there. We also say in this case that the CRN *stably decides* the set $\phi^{-1}(Y) \subseteq \mathbb{R}_{\geq 0}^k$ of inputs that map to output Y . What is the class of sets $S \subseteq \mathbb{R}_{\geq 0}^k$ that are stably decidable by this definition?

For discrete CRNs, the question of “how long” a system takes to stably compute a function has received much interest [1, 3, 25, 28]. In general, asking questions of time-complexity of continuous computation seems more difficult than in the discrete setting. For general polynomial ODEs, the breakthrough work of Bournez, Graça, and Pouly [12] established a surprisingly tight connection between the length of the trajectory and the Turing machine computation time. Closer to our domain of interest, prior work has studied asymptotic convergence speed for the composition of simple CRN motifs as a function of the number of (feedforward) layers [46]. Although not explicitly stated in terms of rate-independent computation, these modules compute in the rate-independent manner as studied here. It is interesting to ask whether these techniques could be adopted to our constructions to articulate and help resolve questions of computation time.

APPENDICES

A FORWARD-INVARIANCE OF ABSENT SIPHONS IN MASS-ACTION SYSTEMS

This section gives an alternate proof of the following result used in Section 2.4, originally due to Angeli, De Leenheer, and Sontag [5].

LEMMA 2.19 ([5], PROPOSITION 5.5). *Fix any assignment of positive mass-action rate constants. Let $\Omega \subseteq \Lambda$ be a set of species. Then Ω is a siphon if and only if, for any state \mathbf{c} such that $\Omega \cap [\mathbf{c}] = \emptyset$ and any state \mathbf{d} that is mass-action reachable from \mathbf{c} , $\Omega \cap [\mathbf{d}] = \emptyset$.*

PROOF. Let $C = (\Lambda, R)$ be the CRN, with positive mass-action rate constants assigned.

To see the forward direction, let Ω be a siphon, and let \mathbf{c} be a state such that $\Omega \cap [\mathbf{c}] = \emptyset$. Consider the reduced CRN $C_{\text{red}} = (\Lambda_{\text{red}}, R_{\text{red}})$ where we remove all species in Ω (i.e., $\Lambda_{\text{red}} = \Lambda \setminus \Omega$) and all reactions referencing them (i.e., $R_{\text{red}} = \{ \langle \mathbf{r}, \mathbf{p} \rangle \in R \mid [\mathbf{r}] \cap \Omega = \emptyset \text{ and } [\mathbf{p}] \cap \Omega = \emptyset \}$). Let $\mathbf{c}_{\text{red}} = \mathbf{c} \upharpoonright \Lambda_{\text{red}}$. Let $\boldsymbol{\rho}' : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}^{\Lambda_{\text{red}}}$ be the mass-action trajectory of C_{red} starting at \mathbf{c}_{red} . Define the trajectory $\boldsymbol{\rho} : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}^{\Lambda}$ of C by $\boldsymbol{\rho}_S(t) = \boldsymbol{\rho}'_S(t)$ if $S \in \Lambda_{\text{red}}$ and $\boldsymbol{\rho}_S(t) = 0$ otherwise, i.e., $\boldsymbol{\rho}$ keeps all of Ω at 0 and otherwise follows $\boldsymbol{\rho}'$.

We claim that $\boldsymbol{\rho}$ is a solution to the mass-action ODEs of C . Since all polynomials, such as those defining mass-action ODEs, are locally Lipschitz (have bounded derivatives in some open

set around every point), the Picard-Lindelöf Theorem implies that the mass-action ODEs have a unique solution. Hence, ρ is the *only* solution to the mass-action ODEs of C starting at c . Since $\rho_S(t) = 0$ for all $S \in \Omega$ and $t \geq 0$, this implies that any d mass-action reachable from c obeys $\Omega \cap [d] = \emptyset$.

To show that ρ is a valid solution to the mass-action ODEs of C , we need to check that

$$\frac{d\rho}{dt} = M \cdot A(\rho(t)) \quad (\text{A.1})$$

for all times $t \geq 0$, where $A(\rho(t))$ is the vector of reaction rates at time $t \geq 0$ (Section 2.4) and M is the stoichiometry matrix (Section 2.2) converting reaction rates to species derivatives. First, let us show that $A_\alpha(\rho(t)) = 0$ for all $t \geq 0$ and $\alpha \notin R_{\text{red}}$. If $\alpha \notin R_{\text{red}}$, then there is some species in Ω that is either a reactant or product of α , and by the fact that Ω is a siphon, we know that there must necessarily be a reactant S of α that is in Ω . Because $\rho_S(t) = 0$ for all $t \geq 0$ by definition and because for mass action ODEs $A_\alpha(\rho(t)) = 0$ if $\rho_S(t) = 0$ for any reactant S of α , this shows that $A_\alpha(\rho(t)) = 0$ for all $t \geq 0$ and $\alpha \notin R_{\text{red}}$. Thus, the right-hand side of Equation (A.1) consists entirely of the contributions of reactions in C_{red} ,

$$M \cdot A(\rho(t)) = \sum_{\alpha \in R_{\text{red}}} M_\alpha A_\alpha(\rho(t)).$$

From this and the fact that ρ' is a solution of the mass-action ODEs of C_{red} it follows that ρ is a solution of the mass-action ODEs for C .

To see the reverse direction, let $\alpha = \langle r, p \rangle$ be a reaction with a product $S \in \Omega$; it suffices to show that α has a reactant in Ω . If S itself is a reactant in α , then we are done, so assume otherwise; then α produces S .

Let c be a state with $[c] = \Lambda_{\text{red}}$, i.e., exactly species not in Ω are present. In particular, $c(S) = 0$. We claim that $A_\alpha(c) = 0$, i.e., α has rate 0 in c . To see why, for the sake of contradiction, suppose $A_\alpha(c) > 0$. To have $d(S) = 0$ for all d mass-action reachable from c , S as a function of time is the constant 0, so $dS/dt = 0$ in c . Then, to balance α 's production of S in c to maintain $dS/dt = 0$, there must be some other reaction β with $M(S, \beta) < 0$ (so β consumes S) and $A_\beta(c) > 0$. Since β consumes S , we know that S is a reactant in β . But, since $c(S) = 0$, the rate of any reaction consuming S is 0 in c , a contradiction. Thus, $A_\alpha(c) = 0$.

Since $[c] = \Lambda_{\text{red}}$ (all species outside of Ω are present), to have mass-action rate 0 in c , α must have at least one reactant in Ω (recall reaction rate constants in a mass-action system are strictly positive). So, Ω is a siphon. \square

The definition of a valid rate schedule (Definition 2.22, part (2)) requires that if some reactant is 0, then the reaction rate is 0. However, the converse implication (if a reaction rate is 0, then some reactant must be 0) holds for mass-action but not more general rate schedules such as segment-reachability, which are allowed to “starve” applicable reactions by holding their rates at 0. The reverse direction of the proof of Lemma 2.19 uses this converse implication, but the forward direction uses only the more general implication of Definition 2.22, part (2). For the forward direction, the key property used from mass-action is its determinism: It has unique solutions, so to show that the siphon remains absent in all possible reachable states it suffices to show that there is just one solution where the siphon remains absent.

B MAX-MIN REPRESENTATION OF CONTINUOUS PIECEWISE LINEAR FUNCTIONS

Here, we prove a slight generalization of Ovchinnikov’s theorem [42]. In Ovchinnikov’s original paper, he only considers piecewise affine functions (in Ovchinnikov’s terminology, piecewise “linear” functions) that are defined on closed domains (that is, closures of open subsets of \mathbb{R}^n).

However, the key proof techniques of Reference [42] did not crucially use this fact. In fact, we apply Theorem 5.11 on non-closed domains such as the sets D_U in the proof of Lemma 5.16. For completeness, we prove the variant of the theorem not requiring D to be closed.

THEOREM 5.11 ([42], THEOREM 2.1). *Let $D \subseteq \mathbb{R}^k$ be convex. For every continuous piecewise affine function $f : D \rightarrow \mathbb{R}$ with components g_1, \dots, g_p , there exists a family $S_1, \dots, S_q \subseteq \{1, \dots, p\}$ such that, for all $\mathbf{x} \in D$, $f(\mathbf{x}) = \max_{i \in \{1, \dots, q\}} \min_{j \in S_i} g_j(\mathbf{x})$.*

To prove the theorem, we first prove three lemmas. The first technical lemma is implicit in Reference [42]. The second and third lemmas correspond to Lemmas 2.1 and 2.2 of Reference [42]. The proofs we give of the second and third lemmas are almost identical in content to the proofs of the corresponding lemmas in Reference [42], except for the fact that we consider piecewise affine functions defined over more general subsets of \mathbb{R}^n . The same is true for our proof of Theorem 5.11, which is again almost identical to the proof of Theorem 2.1 in Reference [42].

LEMMA B.1. *If $f : [a, b] \rightarrow \mathbb{R}$ is a continuous piecewise affine function with components $\{g_1, \dots, g_n\}$, then there are finitely many numbers $a = x_0 < x_1 < \dots < x_m = b$ such that f is affine on $[x_k, x_{k+1}]$ for all k .*

PROOF. Without loss of generality, we can assume that all of the component functions g_i are distinct affine functions. For each i between 1 and n , let D_i be the set of $x \in [a, b]$ such that $g_i(x) = f(x)$. For each i , both g_i and f are continuous, so D_i is closed. Let S be the subset of $[a, b]$ consisting of points $x \in [a, b]$ where x is a member of more than one D_i . For each pair $i \neq j$, we know that g_i and g_j are distinct affine functions, so there can be at most one $x \in [a, b]$ such that $g_i(x) = g_j(x)$. This implies that D_i and D_j can intersect in at most one point, so S must be a finite set. Let $x_0 \dots x_m$ be the elements of $S \cup \{a, b\}$.

Now for a given k write I for the interval (x_k, x_{k+1}) and consider the restriction of f to I . Pick a random point $c \in I$ and suppose $f(c) = g_l(c)$. Then clearly $D_l \cap I$ is nonempty. Because D_l is closed in $[a, b]$, by definition $D_l \cap I$ is closed relative to I . Because $S \cap I = \emptyset$, we also know that

$$D_l \cap I = I \setminus \left(\bigcup_{i \neq l} D_i \right),$$

so $D_l \cap I$ is open relative to I . But the only subset of an interval that is both open and closed is the whole interval, so $I \subseteq D_l$. Therefore, $f = g_l$ when restricted to I , and by continuity, we see that f is affine on the closure of I as well. \square

Note that we define piecewise affine functions to have only finitely many components—without this assumption, the above lemma is false.

LEMMA B.2. *Let $f : [a, b] \rightarrow \mathbb{R}$ be a continuous piecewise affine function. Let $\{g_1, \dots, g_n\}$ be its set of components. Then there is some k such that*

$$g_k(a) \leq f(a) \text{ and } g_k(b) \geq f(b).$$

PROOF. We will first prove the result for $f(a) = f(b) = 0$ and then show how this implies the general case. Given this assumption, if one of the g_i is the zero function, then we are done. If not, since all of the component functions g_i are affine, each g_i can have at most one zero. Since f has finitely many components, this implies that f has finitely many zeros. Let c be the smallest number such that $c > a$ and $f(c) = 0$.

By Lemma B.1, we know that there are some x and y with $a < x < y < c$ and component functions g_k and g_l such that $f = g_k$ on $[a, x]$ and $f = g_l$ on $[y, c]$. If the slope of either g_k or g_l is

non-negative, then we are done:

$$\begin{aligned} g_k(a) &\leq g_k(c) = 0 = f(a) \\ g_k(b) &\geq g_k(c) = 0 = f(b) \end{aligned}$$

and similarly for g_l . But g_k and g_l cannot both have negative slope, for then $f(x) = g_k(x) < 0$ and $f(y) = g_l(y) > 0$, so by the intermediate value theorem there would be some z between x and y such that $f(z) = 0$. This contradicts our assumption that c was the smallest number with $c > a$ and $f(c) = 0$. This concludes the proof assuming that $f(a) = f(b) = 0$.

To deduce the result for a general continuous piecewise affine function from this special case, subtract the affine function

$$\ell(x) = f(a) + f(b) \frac{x - a}{b - a}$$

from f and all of its components. □

LEMMA B.3. *Let D be a convex subset of \mathbb{R}^n and let $f : D \rightarrow \mathbb{R}$ be a continuous piecewise affine function. If the components of f are $\{g_1 \dots g_n\}$, then for every pair of vectors \mathbf{a} and \mathbf{b} in D , there is some k such that*

$$g_k(\mathbf{a}) \leq f(\mathbf{a}) \text{ and } g_k(\mathbf{b}) \geq f(\mathbf{b}).$$

PROOF. Because D is convex, the straight-line interval between \mathbf{a} and \mathbf{b} is contained in D . Apply Lemma B.2 to the restriction of f to this interval. □

Finally, we are ready to prove Theorem 5.11.

PROOF OF THEOREM 5.11. For each $\mathbf{b} \in D$, define the set $S_{\mathbf{b}} \subseteq \{1 \dots p\}$ as

$$S_{\mathbf{b}} = \{i \mid g_i(\mathbf{b}) \geq f(\mathbf{b})\}.$$

Let

$$F_{\mathbf{b}}(\mathbf{x}) = \min_{i \in S_{\mathbf{b}}} g_i(\mathbf{x}).$$

Because there is always some component function g_j with $g_j(\mathbf{b}) = f(\mathbf{b})$, we see that $F_{\mathbf{b}}(\mathbf{b}) = f(\mathbf{b})$ for every $\mathbf{b} \in D$. Also, by Lemma B.3, we know that for every $\mathbf{a} \in D$, there is some component function $g_k \in S_{\mathbf{b}}$ with $g_k(\mathbf{a}) \leq f(\mathbf{a})$, so $F_{\mathbf{b}}(\mathbf{a}) \leq f(\mathbf{a})$ for every pair $\mathbf{b}, \mathbf{a} \in D$. This implies that

$$f(\mathbf{x}) = \max_{\mathbf{b} \in D} F_{\mathbf{b}}(\mathbf{x}) = \max_{\mathbf{b} \in D} \min_{i \in S_{\mathbf{b}}} g_i(\mathbf{x}). \quad (\text{B.2})$$

Since $\{1 \dots p\}$ is a finite set, it has only finitely many subsets, so each $S_{\mathbf{b}}$ is equal to one of finitely many sets S_j . We can therefore replace the maximum over all $\mathbf{b} \in D$ in Equation (B.2) with a maximum over finitely many functions. □

C FINDING RATIONAL SOLUTIONS TO SYSTEMS OF LINEAR EQUATIONS

It is well-known that a system of linear equations with rational coefficients has a rational solution if and only if it has a real solution. The following result shows the slightly generalized claim that rational solutions exist arbitrarily close to all real solutions (i.e., the rational solutions are dense in the real solutions).

LEMMA C.1. *Let $A\mathbf{x} = \mathbf{b}$ be a system of linear equations, where A is a matrix with rational coefficients and \mathbf{b} is a vector with rational coefficients. If the equation has a solution \mathbf{x} with real coefficients, then for any $\varepsilon > 0$, it has a solution with \mathbf{x}' rational coefficients such that $\|\mathbf{x}' - \mathbf{x}\| < \varepsilon$.*

PROOF. Let n be the number of rows of A and the length of \mathbf{b} . Let m be the number of columns of A and the length of \mathbf{x} . Because A has rational entries, using elementary row and column operations it can be decomposed as PNQ where P is an $m \times m$ invertible rational matrix, Q is an $n \times n$ invertible rational matrix, and

$$N_{ij} = \begin{cases} 1 & i = j \text{ and } i \leq r \\ 0 & \text{otherwise,} \end{cases}$$

where r is the rank of M . Let $\mathbf{y} = Q\mathbf{x}$ and let $\mathbf{c} = P^{-1}\mathbf{b}$, so $N\mathbf{y} = \mathbf{c}$. If $r = n$, then all of the entries of \mathbf{y} must be rational, since $\mathbf{y}_i = \mathbf{c}_i$ for all i and all of the entries of \mathbf{c}_i are rational. Then the entries of \mathbf{x} must also all be rational, since $\mathbf{x} = Q^{-1}\mathbf{y}$ and Q has all rational entries. As a result, if $r = n$, then we can just take $\mathbf{x}' = \mathbf{x}$.

However, if $r < n$, then let

$$\delta = \frac{\varepsilon}{\sqrt{n-r}\|Q^{-1}\|},$$

where

$$\|Q^{-1}\| = \sup_{\mathbf{v} \neq 0} \frac{\|Q^{-1}\mathbf{v}\|}{\|\mathbf{v}\|}$$

is the operator norm of $\|Q^{-1}\|$. Now let \mathbf{y}' be a vector such that $\mathbf{y}'_i = \mathbf{y}_i$ for $i \leq r$ and \mathbf{y}'_i is a rational number such that $|\mathbf{y}'_i - \mathbf{y}_i| < \delta$ for $i > r$. All of the components of \mathbf{y}' are rational: \mathbf{y}'_i is rational by construction for $i > r$, and $\mathbf{y}'_i = \mathbf{y}_i = \mathbf{c}_i$ is rational for $i \leq r$. Moreover, the fact that $\mathbf{y}'_i = \mathbf{c}_i$ for $i \leq r$ shows that $N\mathbf{y}' = \mathbf{c}$.

If we take $\mathbf{x}' = Q^{-1}\mathbf{y}'$, then all of the components of \mathbf{x}' are rational, and $M\mathbf{x}' = \mathbf{b}$, since

$$M\mathbf{x}' = PNQ\mathbf{x}' = PN\mathbf{y}' = P\mathbf{c} = \mathbf{b}.$$

Finally, we know that $\|\mathbf{x}' - \mathbf{x}\| < \varepsilon$, since

$$\begin{aligned} \|\mathbf{x}' - \mathbf{x}\| &= \|Q^{-1}(\mathbf{y}' - \mathbf{y})\| \\ &\leq \|Q^{-1}\| \cdot \|\mathbf{y}' - \mathbf{y}\| \\ &= \|Q^{-1}\| \sqrt{\sum_{i=1}^n (\mathbf{y}'_i - \mathbf{y}_i)^2} \\ &< \|Q^{-1}\| \sqrt{(n-r)\delta^2} \\ &= \varepsilon. \end{aligned}$$

This shows that \mathbf{x}' is our desired solution. □

D BOUNDING REACTION FLUXES IN STRAIGHT-LINE REACHABILITY

This section is devoted to proving Lemma D.4. Intuitively, it shows that if a CRN can reach from state \mathbf{c} to state \mathbf{d} by a straight line (of length $\|\mathbf{d} - \mathbf{c}\|$), then the reaction fluxes required can be bounded by $O(\|\mathbf{d} - \mathbf{c}\|)$. This is nontrivial, since one can have reactions that cancel, e.g., $X \rightarrow Y$ and $Y \rightarrow X$. The same straight line from \mathbf{c} to \mathbf{d} could result from arbitrarily large but equal fluxes of each reaction (plus some other reactions). Lemma D.4 states that we never *need* arbitrarily large fluxes to get from \mathbf{c} to \mathbf{d} .

Definition D.1. A convex polyhedral cone C is a subset of a vector space V such that there exist vectors $v_1, \dots, v_k \in V$ so C is exactly the set of $x \in V$ that can be written as $x = \sum_i \lambda_i v_i$ with all $\lambda_i \geq 0$. Such a set $S = \{v_1 \dots v_k\}$ is called a *spanning set* for C , and we say that C is *spanned by* S . Given a set $S = \{v_1 \dots v_k\}$ of vectors in V , we write C_S for the convex polyhedral cone spanned by S .

LEMMA D.2. *Let C be a convex cone with spanning set $S = \{v_1 \dots v_k\}$. Let \mathcal{I} be the collection of all linearly independent subsets of S . Then*

$$C = \bigcup_{S' \in \mathcal{I}} C_{S'}.$$

PROOF. Suppose to the contrary that there was some $x \in C$ not contained in $C_{S'}$ for any $S' \in \mathcal{I}$. Let T be a minimal subset of S such that $x \in C_T$. Then, since $x \in C_T$, we can write x as

$$x = \sum_{v_i \in T} \lambda_i v_i.$$

To produce a contradiction, let us show that we can express x as

$$x = \sum_{v_i \in T} \lambda'_i v_i$$

with some $\lambda'_i = 0$. This will imply that $x \in C_{T \setminus \{v_i\}}$, contradicting the minimality of T . By assumption, since $x \in T$, we know that $T \notin \mathcal{I}$, so there is some nontrivial linear relationship

$$\sum_{v_i \in T} \mu_i v_i = 0$$

among the $v_i \in T$. By negating all of the μ_i if needed, we can assume that at least one $\mu_i < 0$. Let C be the constant

$$C = \min_{\mu_i < 0} \frac{-\lambda_i}{\mu_i}.$$

Let us show that $\lambda_j + C\mu_j \geq 0$ for all j . Because $\lambda_i > 0$ for all i , we know that $C > 0$. As a result, if $\mu_j \geq 0$, then necessarily $\lambda_j + C\mu_j \geq 0$. However, if $\mu_j < 0$, then $C \leq -\lambda_j/\mu_j$, so

$$\lambda_j + C\mu_j \geq \lambda_j + \left(\frac{\lambda_j}{\mu_j}\right)\mu_j = 0.$$

Moreover, for some $\mu_{i_0} < 0$, we know that $\lambda_{i_0} + C\mu_{i_0} = 0$. As a result,

$$x = \left(\sum_{v_i \in T} \lambda_i v_i \right) + C \left(\sum_{v_i \in T} \mu_i v_i \right) = \sum_{v_i \in T} (\lambda_i + C\mu_i) v_i = \sum_{v_i \in T} \lambda'_i v_i,$$

where $\lambda'_{i_0} = 0$. Thus, $x \in C_{T \setminus \{v_{i_0}\}}$, contradicting the minimality of T and therefore the existence of x . \square

LEMMA D.3. *Let C be a convex cone with spanning set $S = \{v_1 \dots v_k\}$. Then there is a constant K depending only on S so for any $x \in C$, there is some representation of x as*

$$x = \sum_i \lambda_i v_i$$

with $0 \leq \lambda_i \leq K\|x\|$. Additionally, if there is some subset $T \subseteq S$ such that $x \in C_T$, then the above representation can be chosen with $\lambda_i = 0$ for any $v_i \notin T$.

PROOF. Let \mathcal{I} be the collection of linearly independent subsets of S . If $S' \in \mathcal{I}$ and $x \in C_{S'}$, then because the vectors $v_1 \dots v_k$ in S' are linearly independent, there is a unique way to write

$$x = \sum_{v_i \in S'} \lambda_i v_i$$

and by Reference [33, Lemma 2.4-1], we know there is some constant $K_{S'}$ so $0 \leq \lambda_i \leq K_{S'} \|x\|$. Take $K = \max_{S' \in \mathcal{I}} K_{S'}$. Applying Lemma D.2 to C_T , we can find some linearly independent collection of vectors T' such that $T' \subseteq T$ and $x \in C_{T'}$. Then

$$x = \sum_{v_i \in T'} \lambda_i v_i$$

with $0 \leq \lambda_i \leq K_{T'} \|x\|$. Because $T' \subseteq T \subseteq S$, we know that $T' \in \mathcal{I}$, so $K_{T'} \leq K$, and, since $T' \subseteq T$, we know that above sum only ranges over vectors $v_i \in T$. \square

Finally, we have the main result of this section.

LEMMA D.4. *Fix a CRN C and suppose a flux vector $\mathbf{u} \in \mathbb{R}_{\geq 0}^R$ is applicable at a state $\mathbf{c} \in \mathbb{R}_{\geq 0}^\Lambda$. Let $\ell = \|\mathbf{Mu}\|$ be the length of the straight-line segment in \mathbb{R}^Λ given by \mathbf{Mu} . Then there exists a constant K , depending only on C and independent of \mathbf{u} and \mathbf{c} , such that there exists a flux vector \mathbf{u}' where \mathbf{u}' is also applicable at \mathbf{c} , the length of \mathbf{u}' is bounded as $\|\mathbf{u}'\| \leq K\ell$, and $\mathbf{Mu} = \mathbf{Mu}'$.*

PROOF. Let $\{v_1 \dots v_m\}$ be the standard basis vectors of \mathbb{R}^R . Because $\mathbf{u} = \sum_i \lambda_i v_i$ with all $\lambda_i \geq 0$, we know that \mathbf{Mu} is contained in the convex polyhedral cone spanned by $\{Mv_1 \dots Mv_m\}$. By Lemma D.3, we know that there are λ'_i such that $\mathbf{Mu} = \sum_i \lambda'_i Mv_i$ and $0 \leq \lambda'_i \leq K\|\mathbf{Mu}\|$, and moreover $\lambda'_i = 0$ whenever $\lambda_i = 0$. Let $\mathbf{u}' = \sum_i \lambda'_i v_i$. Then \mathbf{u}' is still applicable at \mathbf{c} because $\lambda'_i > 0$ implies $\lambda_i > 0$ and we assumed that \mathbf{u} was applicable at \mathbf{c} . Also, note that

$$\mathbf{Mu}' = \sum_i \lambda'_i Mv_i = \mathbf{Mu}.$$

Finally, we know that

$$\|\mathbf{u}'\| \leq \sum_i \lambda'_i \leq \sum_i K\|\mathbf{Mu}\| \leq mK\ell,$$

so \mathbf{u}' is our desired vector in $\mathbb{R}_{\geq 0}^R$. \square

E PARTIAL STATES AND REACHABILITY

In this section, we define a notion of “partial” states and reachability, which are used in the proof of Theorem 3.3. If $\Delta \subseteq \Lambda$, then we say $\mathbf{p} \in \mathbb{R}_{\geq 0}^\Delta$ is a *partial state*. Given a state $\mathbf{c} \in \mathbb{R}_{\geq 0}^\Lambda$, recall that $\mathbf{c} \upharpoonright \Delta$ is \mathbf{c} restricted to Δ , i.e., the partial state $\mathbf{p} = \mathbf{c} \upharpoonright \Delta \in \mathbb{R}_{\geq 0}^\Delta$ such that $\mathbf{p}(S) = \mathbf{c}(S)$ for all $S \in \Delta$.

Let $k \in \mathbb{N} \cup \{\infty\}$. Given a state $\mathbf{c} \in \mathbb{R}_{\geq 0}^\Lambda$ and a partial state $\mathbf{p} \in \mathbb{R}_{\geq 0}^\Delta$, we write $\mathbf{c} \rightsquigarrow^k \mathbf{p}$ if there is a sequence of states $\mathbf{b}_0, \dots, \mathbf{b}_k \in \mathbb{R}_{\geq 0}^\Lambda$ such that $\mathbf{c} = \mathbf{b}_0 \rightarrow^1 \mathbf{b}_1 \rightarrow^1 \mathbf{b}_2 \rightarrow^1 \dots \rightarrow^1 \mathbf{b}_k$, with $\mathbf{p} = \mathbf{b}_k \upharpoonright \Delta$ if $k \in \mathbb{N}$, or $\mathbf{p} = \lim_{i \rightarrow \infty} (\mathbf{b}_i \upharpoonright \Delta)$ if $k = \infty$. We write $\mathbf{c} \rightsquigarrow \mathbf{p}$ *via* $(\mathbf{b}_i)_{i=1}^k$ if $\mathbf{c} \rightsquigarrow^k \mathbf{p}$ for some $k \in \mathbb{N} \cup \{\infty\}$ with intermediate states $\mathbf{b}_0, \mathbf{b}_1, \dots$ as above, or simply $\mathbf{c} \rightsquigarrow \mathbf{p}$ when the intermediate states $\mathbf{b}_0, \mathbf{b}_1, \dots$ are implicit. We write $\mathbf{c} \rightsquigarrow_{ss}^\infty \mathbf{p}$ *via* $(\mathbf{b}_i)_{i=1}^k$ if $\mathbf{c} = \mathbf{b}'_0 \rightarrow^1 \mathbf{b}'_1 \rightarrow^1 \dots$, where for some subsequence $\mathbf{b}_1, \mathbf{b}_2, \dots$ of $\mathbf{b}'_1, \mathbf{b}'_2, \dots$, we have $\mathbf{p} = \lim_{i \rightarrow \infty} (\mathbf{b}_i \upharpoonright \Delta)$, i.e., an infinite subsequence of states converges on concentrations in Δ .

Note that if there is a state \mathbf{d} such that $\mathbf{c} \rightsquigarrow \mathbf{d}$ and $\mathbf{d} \upharpoonright \Delta = \mathbf{p}$, then $\mathbf{c} \rightsquigarrow \mathbf{p}$, but it is not apparent from the definition that this is the only way for a partial state to be reachable if the number of line segments is infinite. In particular, it could be that the sequence $\mathbf{b}_0, \mathbf{b}_1, \dots$ does not converge to any state (even though the partial states $\mathbf{b}_0 \upharpoonright \Delta, \mathbf{b}_1 \upharpoonright \Delta, \dots$ converge to \mathbf{p}), if some concentration values outside of Δ do not converge (for instance, they may oscillate or go to infinity).

Our goal now is to show that in fact, if a partial state \mathbf{p} is reachable (or even merely $\rightsquigarrow_{ss}^\infty$ reachable), then there is a particular state reachable whose restriction to Δ is \mathbf{p} .

THEOREM E.1. *Let $\Delta \subseteq \Lambda$, let $\mathbf{c} \in \mathbb{R}_{\geq 0}^\Lambda$ be a state, and $\mathbf{p} \in \mathbb{R}_{\geq 0}^\Delta$ be a partial state. If $\mathbf{c} \rightsquigarrow_{ss}^\infty \mathbf{p}$, then there is a state \mathbf{d} such that $\mathbf{c} \rightsquigarrow \mathbf{d}$ and $\mathbf{p} = \mathbf{d} \upharpoonright \Delta$.*

Furthermore, if $\mathbf{c} \rightsquigarrow_{ss}^\infty \mathbf{p}$ via $(\mathbf{b}_i)_{i=1}^k$ for $k \in \mathbb{N} \cup \{\infty\}$, then there is a partition of Λ into Λ_b and Λ_u , with $\Delta \subseteq \Lambda_b$, and a subsequence $(\mathbf{r}_i)_i$ of $(\mathbf{b}_i)_i$ so $\lim_{i \rightarrow \infty} \mathbf{r}_i(S) = \infty$ for all $S \in \Lambda_u$, and $\lim_{i \rightarrow \infty} \mathbf{r}_i(S) = \mathbf{d}(S)$ for all $S \in \Lambda_b$.

PROOF. The finite case is immediate from the definition of \rightsquigarrow for partial states (choose $\mathbf{d} = \mathbf{b}_k$ and $\Lambda_u = \emptyset$), so assume $\mathbf{c} \rightsquigarrow_{ss}^\infty \mathbf{p}$. Then there is an infinite sequence of states (the converging subsequence in the definition of $\rightsquigarrow_{ss}^\infty$) $\mathbf{b}_0, \mathbf{b}_1, \dots \in \mathbb{R}_{\geq 0}^\Lambda$ such that $\mathbf{c} = \mathbf{b}_0$, each $\mathbf{b}_i \rightsquigarrow^{k_i} \mathbf{b}_{i+1}$ for some $k_i \in \mathbb{N}$, and $\mathbf{p} = \lim_{i \rightarrow \infty} \mathbf{b}_i \upharpoonright \Delta$.

Let $\Gamma = \Lambda \setminus \Delta$ be the species outside of Δ , which may not be converging in the subsequence $(\mathbf{b}_i)_i$. Intuitively, our goal will be to make some species in Γ converge and all others simultaneously diverge to infinity. Then, by driving the concentrations of the diverging species sufficiently large, then removing them from the system, we obtain a reduced CRN where all species converge to a single state. We apply Corollary 2.15 to this reduced CRN to find a finite path of $m + 1$ segments reaching this state. Finally, we argue that this path is applicable even in the original CRN, because the species that were removed first had their concentrations driven large enough that the finite path does not have sufficient flux to send any of them to 0. Thus, at the end of this finite path, all species converge to some concentration.

Formally, define the set Γ_u (the “unbounded” species in Γ) iteratively as follows: If there is any $S_1 \in \Gamma$ such that $\limsup_{i \rightarrow \infty} \mathbf{b}_i(S_1) = \infty$, then put S_1 in Γ_u , otherwise Γ_u is defined to be \emptyset . Then pick a subsequence $\mathbf{b}'_0, \mathbf{b}'_1, \dots$ of $\mathbf{b}_0, \mathbf{b}_1, \dots$ such that, for all $i \in \mathbb{N}$, $\mathbf{b}'_i(S_1) \geq i$; such a subsequence exists, since $\limsup_{i \rightarrow \infty} \mathbf{b}_i(S_1) = \infty$. Now, from *that* subsequence, if any species S_2 obeys $\limsup_{i \rightarrow \infty} \mathbf{b}'_i(S_2) = \infty$, then place S_2 in Γ_u and choose a subsequence $\mathbf{b}''_0, \mathbf{b}''_1, \dots$ of $\mathbf{b}'_0, \mathbf{b}'_1, \dots$ where $\mathbf{b}''_i(S_2) \geq i$ for each $i \in \mathbb{N}$; note that because $\mathbf{b}''_i = \mathbf{b}'_{i'}$ for $i' \geq i$, we also have $\mathbf{b}''_i(S_1) \geq i$.

Repeat this alternation of choosing a species S_j to put in Γ_u and picking a subsequence, until in the final subsequence $\mathbf{b}'''_0, \mathbf{b}'''_1, \dots$ every remaining species S obeys $\limsup_{i \rightarrow \infty} \mathbf{b}'''_i(S) < \infty$. Define the set $\Gamma_b = \Gamma \setminus \Gamma_u$; by construction each species in Γ_b has bounded concentrations in $\mathbf{b}'''_0, \mathbf{b}'''_1, \dots$ (Though some may have had unbounded concentrations in the original sequence $\mathbf{b}_0, \mathbf{b}_1, \dots$)

Now, since concentrations of species in Γ_b are in a closed, bounded (i.e., compact) set, there is a subsequence $\mathbf{r}_0, \mathbf{r}_1, \dots$ of $\mathbf{b}'''_0, \mathbf{b}'''_1, \dots$ that converges on concentrations for species in Γ_b . Furthermore, all subsequences of $\mathbf{b}_0, \mathbf{b}_1, \dots$ we have taken so far can be assumed without loss of generality to contain \mathbf{b}_0 . Thus, the subsequence $\mathbf{r}_0, \mathbf{r}_1, \dots$ obeys

- (1) $\mathbf{r}_0 = \mathbf{b}_0 (= \mathbf{c})$,
- (2) for all $S \in \Gamma_u$ and $i \in \mathbb{N}$, $\mathbf{r}_i(S) \geq i$ (concentrations in Γ_u increase to infinity simultaneously), and
- (3) for all $S \in \Gamma_b \cup \Delta$, $\lim_{i \rightarrow \infty} \mathbf{r}_i(S)$ exists and is finite (all other concentrations converge).

Consider the “quotient” CRN $C_q = (\Lambda_q, R_q)$, where $\Lambda_q = \Lambda \setminus \Gamma_u = \Delta \cup \Gamma_b$, and R_q is defined by taking each reaction from R and removing any species from it in Γ_u . For example, if $A, B \in \Gamma_u$ and $C, D, E \notin \Gamma_u$, then the reaction $A + B + 2C \rightarrow 2A + D + E$ becomes $2C \rightarrow D + E$. Then for each state $\mathbf{r} \in \mathbb{R}_{\geq 0}^{\Lambda_q}$ of the original CRN C , its partial state $\mathbf{r} \upharpoonright \Lambda_q$ is a (normal) state of C_q .

For each $i \in \mathbb{N}$, let $\mathbf{q}_i = \mathbf{r}_i \upharpoonright \Lambda_q$ be the state of C_q corresponding to \mathbf{r}_i . Note that $\mathbf{q}_0 = \mathbf{c} \upharpoonright \Lambda_q$. By the definition of $\Lambda_q = \Delta \cup \Gamma_b$ and the convergence of concentrations in both Δ (by the definition of $\mathbf{c} \rightsquigarrow \mathbf{p}$) and Γ_b (shown as part (3) above), the sequence $\mathbf{q}_0, \mathbf{q}_1, \dots$ converges to some state $\mathbf{q} \in \mathbb{R}_{\geq 0}^{\Lambda_q}$. This implies that $\mathbf{q}_i \rightsquigarrow_{ss}^\infty \mathbf{q}$ for each $i \in \mathbb{N}$. Note that $\mathbf{q} \upharpoonright \Delta = \mathbf{p}$.

Since the \mathbf{q}_i ’s converge to \mathbf{q} , there is some i_0 such that, for all $i \geq i_0$, $\|\mathbf{q} - \mathbf{q}_i\| \leq 1$. Since each $\mathbf{q}_i \rightsquigarrow_{ss}^\infty \mathbf{q}$, by Theorem 2.14, $\mathbf{q}_i \rightsquigarrow^{m+1} \mathbf{q}$, where $m = \min\{|\Lambda_q|, |R_q|\}$, and for some constant K depending only on C_q , the total reaction fluxes do not exceed K along the entire path of $m + 1$ segments.

Let $C = \max\{\mathbf{r}'(S) \mid S \in \Gamma_u, \langle \mathbf{r}', \mathbf{p}' \rangle \in R\}$ be the maximum reactant coefficient of any reactant in Γ_u for any reaction in R . Choose $i_1 = \max\{C \cdot K, i_0\}$. Then $\mathbf{q}_{i_1} \rightsquigarrow \mathbf{q}$ with total reaction flux at most K , since $i_1 \geq i_0$.

Consider running the same reaction fluxes on the original CRN, from the state $\mathbf{b}_{i'}$, choosing i' such that $\mathbf{b}_{i'} \upharpoonright \Lambda_q = \mathbf{q}_{i_1}$. By property (2) above, there is sufficient concentration $C \cdot K$ of each reactant in Γ_u in state $\mathbf{b}_{i'}$ for these reactions to remain applicable along the entire path. Let $\mathbf{d} \in \mathbb{R}_{\geq 0}^\Lambda$ be the state reached at the end of this path, then $\mathbf{b}_{i'} \rightsquigarrow \mathbf{d}$. Note that $\mathbf{d} \upharpoonright \Lambda_q = \mathbf{q}$, and recall $\mathbf{q} \upharpoonright \Delta = \mathbf{p}$, so $\mathbf{d} \upharpoonright \Delta = \mathbf{p}$. Since $\mathbf{c} \rightsquigarrow \mathbf{b}_{i'}$, by transitivity of \rightsquigarrow , we have $\mathbf{c} \rightsquigarrow \mathbf{d}$, proving the theorem. \square

Note that by Corollary 2.15, if $\mathbf{c} \rightsquigarrow \mathbf{p}$ for a partial state \mathbf{p} , then $\mathbf{c} \rightsquigarrow^{m+1} \mathbf{p}$, where $m = \min\{|\Lambda|, |R|\}$.

ACKNOWLEDGMENTS

We thank Manoj Gopalkrishnan, Elisa Franco, Damien Woods, and the organizers and participants of the American Mathematical Institute workshop on Mathematical Problems Arising from Biochemical Reaction Networks for insightful discussions. We are grateful to anonymous reviewers for insightful comments and suggestions that have greatly improved this article.

REFERENCES

- [1] Dan Alistarh, James Aspnes, David Eisenstat, Rati Gelashvili, and Ronald L. Rivest. 2017. Time-space trade-offs in population protocols. In *28th Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM, 2560–2579.
- [2] Dan Alistarh, James Aspnes, and Rati Gelashvili. 2018. Space-optimal majority in population protocols. In *29th Annual ACM-SIAM Symposium on Discrete Algorithms*. 2221–2239.
- [3] Dan Alistarh and Rati Gelashvili. 2018. Recent algorithmic advances in population protocols. *ACM SIGACT News* 49, 3 (2018), 63–73.
- [4] David Angeli, Patrick De Leenheer, and Eduardo D. Sontag. 2006. On the structural monotonicity of chemical reaction networks. In *45th IEEE Conference on Decision and Control*. IEEE, 7–12.
- [5] David Angeli, Patrick De Leenheer, and Eduardo D. Sontag. 2007. A Petri net approach to the study of persistence in chemical reaction networks. *Mathem. Biosci.* 210, 2 (2007), 598–618.
- [6] Dana Angluin, James Aspnes, Zoë Diamadi, Michael Fischer, and René Peralta. 2006. Computation in networks of passively mobile finite-state sensors. *Distrib. Comput.* 18, 4 (2006), 235–253. DOI : <http://dx.doi.org/10.1007/s00446-005-0138-3>
- [7] Dana Angluin, James Aspnes, and David Eisenstat. 2006. Stably computable predicates are semilinear. In *PODC 2006: 25th Annual ACM Symposium on Principles of Distributed Computing*. ACM Press, New York, NY, 292–299. DOI : <https://doi.org/10.1145/1146381.1146425>
- [8] Dana Angluin, James Aspnes, and David Eisenstat. 2008. Fast computation by population protocols with a leader. *Distrib. Comput.* 21, 3 (Sept. 2008), 183–199.
- [9] James Aspnes and Eric Ruppert. 2007. An introduction to population protocols. *Bull. Europ. Assoc. Theoret. Comput. Sci.* 93 (2007), 98–117.
- [10] Naama Barkai and Stan Leibler. 1997. Robustness in simple biochemical networks. *Nature* 387, 6636 (1997), 913–917.
- [11] Amanda Belleville, David Doty, and David Soloveichik. 2017. Hardness of computing and approximating predicates and functions with leaderless population protocols. In *44th International Colloquium on Automata, Languages, and Programming (Leibniz International Proceedings in Informatics (LIPIcs))*, Vol. 80. 141:1–141:14.
- [12] Olivier Bournez, Daniel S. Graça, and Amaury Pouly. 2017. Polynomial time corresponds to solutions of polynomial ordinary differential equations of polynomial length. *J. ACM* 64, 6 (Oct. 2017). DOI : <https://doi.org/10.1145/3127496>
- [13] Robert Brijder. 2019. Computing with chemical reaction networks: A tutorial. *Natur. Comput.* 18, 1 (2019), 119–137.
- [14] Luca Cardelli. 2011. Strand algebras for DNA computing. *Natur. Comput.* 10, 1 (2011), 407–428.
- [15] Luca Cardelli and Attila Csikász-Nagy. 2012. The cell cycle switch computes approximate majority. *Sci. Rep.* 2 (2012).
- [16] Adam Case, Jack H. Lutz, and Donald M. Stull. 2018. Reachability problems for continuous chemical reaction networks. *Natur. Comput.* 17, 2 (2018), 223–230.
- [17] Cameron Chalk, Niels Kernerup, Wyatt Reeves, and David Soloveichik. 2021. Composable rate-independent computation in continuous chemical reaction networks. *IEEE/ACM Trans. Computat. Biol. Bioinform.* 18, 1 (2021), 250–260.

- [18] Ho-Lin Chen, David Doty, and David Soloveichik. 2014. Deterministic function computation with chemical reaction networks. *Natur. Comput.* 13, 4 (2014), 517–534. DOI : <https://doi.org/10.1007/s11047-013-9393-6>
- [19] Ho-Lin Chen, David Doty, and David Soloveichik. 2014. Rate-independent computation in continuous chemical reaction networks. In *5th Conference on Innovations in Theoretical Computer Science*.
- [20] Yuan-Jyue Chen, Neil Dalcchau, Niranjan Srinivas, Andrew Phillips, Luca Cardelli, David Soloveichik, and Georg Seelig. 2013. Programmable chemical controllers made from DNA. *Nat. Nanotechnol.* 8, 10 (2013), 755–762.
- [21] Matthew Cook, David Soloveichik, Erik Winfree, and Jehoshua Bruck. 2009. Programmability of chemical reaction networks. In *Algorithmic Bioprocesses*, Anne Condon, David Harel, Joost N. Kok, Arto Salomaa, and Erik Winfree (Eds.). Springer, Berlin, 543–584.
- [22] Wojciech Czerwiński and Łukasz Orlikowski. 2022. Reachability in vector addition systems is Ackermann-complete. In *IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*. 1229–1240.
- [23] Élisabeth Degrand, François Fages, and Sylvain Soliman. 2020. Graphical conditions for rate independence in chemical reaction networks. In *Computational Methods in Systems Biology*, Alessandro Abate, Tatjana Petrov, and Verena Wolf (Eds.). Springer International Publishing, Cham, 61–78.
- [24] David Doty and Eric Severson. 2021. Ppsim: A software package for efficiently simulating and visualizing population protocols. In *19th International Conference on Computational Methods in Systems Biology*. 245–253. Retrieved from <https://arxiv.org/abs/2105.04702>.
- [25] David Doty and David Soloveichik. 2018. Stable leader election in population protocols requires linear time. *Distrib. Comput.* 31, 4 (2018), 257–271.
- [26] François Fages, Steven Gay, and Sylvain Soliman. 2015. Inferring reaction systems from ordinary differential equations. *Theoret. Comput. Sci.* 599 (2015), 64–78. DOI : <https://doi.org/10.1016/j.tcs.2014.07.032>
- [27] François Fages, Guillaume Le Guludec, Olivier Bournez, and Amaury Pouly. 2017. Strong Turing completeness of continuous chemical reaction networks and compilation of mixed analog-digital programs. In *International Conference on Computational Methods in Systems Biology*. Springer, 108–127.
- [28] Leszek Gąsieniec and Grzegorz Staehowiak. 2018. Fast space optimal leader election in population protocols. In *29th Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM, 2653–2667.
- [29] Daniel T. Gillespie. 1977. Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.* 81, 25 (1977), 2340–2361.
- [30] Manoj Gopalkrishnan, Ezra Miller, and Anne Shiu. 2013. A projection argument for differential inclusions, with applications to persistence of mass-action kinetics. *Symm., Integrabil. Geom.: Meth. Applic.* 9, 0 (2013), 25–25.
- [31] Hooman Hashemi, Ben Chugg, and Anne Condon. 2020. Composable computation in leaderless, discrete chemical reaction networks. In *26th International Conference on DNA Computing and Molecular Programming (DNA'20)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik.
- [32] Shigeru Kondo and Takashi Miura. 2010. Reaction-diffusion model as a framework for understanding biological pattern formation. *Science* 329, 5999 (2010), 1616–1620.
- [33] Erwin Kreyszig. 1991. *Introductory Functional Analysis with Applications*, Vol. 17. John Wiley & Sons.
- [34] Thomas G. Kurtz. 1972. The relationship between stochastic and deterministic models for chemical reactions. *J. Chem. Phys.* 57, 7 (1972), 2976–2978.
- [35] James I. Lathrop, Jack H. Lutz, Robyn R. Lutz, Hugh D. Potter, and Matthew R. Riley. 2020. Population-induced phase transitions and the verification of chemical reaction networks. In *26th International Conference on DNA Computing and Molecular Programming (Leibniz International Proceedings in Informatics (LIPIcs))*, Cody Geary and Matthew J. Patitz (Eds.), Vol. 174. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, Dagstuhl, Germany, 5:1–5:17. DOI : <https://doi.org/10.4230/LIPIcs.DNA.2020.5>
- [36] John M. Lee. 2013. *Introduction to Smooth Manifolds* (2nd ed.). Springer.
- [37] Jérôme Leroux. 2022. The reachability problem for Petri nets is not primitive recursive. In *IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*. 1241–1252.
- [38] Richard J. Lipton. 1976. *The Reachability Problem Requires Exponential Space*. Technical Report. Yale University.
- [39] Ernst W. Mayr. 1984. An algorithm for the general Petri net reachability problem. *SIAM J. Comput.* 13, 3 (1984), 441–460.
- [40] James R. Munkres. 2000. *Topology* (2nd ed.). Prentice Hall, Upper Saddle River, NJ, 108–109.
- [41] Jan Mycielski. 1992. Games with perfect information. *Handb. Game Theor. Econ. Applic.* 1 (1992), 41–70.
- [42] Sergei Ovchinnikov. 2002. Max-min representation of piecewise linear functions. *Contrib. Algeb. Geom.* 43, 1 (2002), 297–302.
- [43] Halsey Lawrence Royden and Patrick Fitzpatrick. 1988. *Real Analysis* (4th ed.), Vol. 32. Macmillan New York.
- [44] Sayed Ahmad Salehi, Keshab K. Parhi, and Marc D. Riedel. 2017. Chemical reaction networks for computing polynomials. *ACS Synthet. Biol.* 6, 1 (2017), 76–83.

- [45] Michael S. Samoilov and Adam P. Arkin. 2006. Deviant effects in molecular reaction pathways. *Nat. Biotechnol.* 24, 10 (2006), 1235–1240.
- [46] Georg Seelig and David Soloveichik. 2009. Time-complexity of multilayered DNA strand displacement circuits. In *International Workshop on DNA-based Computers*. Springer, 144–153.
- [47] Phillip Senum and Marc Riedel. 2011. Rate-independent constructs for chemical computation. *PLoS One* 6, 6 (2011), e21414.
- [48] Eric E. Severson, David Haley, and David Doty. 2021. Composable computation in discrete chemical reaction networks. *Distrib. Comput.* 34, 6 (2021), 437–461.
- [49] David Soloveichik, Matthew Cook, Erik Winfree, and Jehoshua Bruck. 2008. Computation with finite stochastic chemical reaction networks. *Natur. Comput.* 7, 4 (2008), 615–633. DOI : <http://dx.doi.org/10.1007/s11047-008-9067-y>
- [50] David Soloveichik, Georg Seelig, and Erik Winfree. 2010. DNA as a universal substrate for chemical kinetics. *Proc. Nat. Acad. Sci.* 107, 12 (2010), 5393.
- [51] Niranjana Srinivas, James Parkin, Georg Seelig, Erik Winfree, and David Soloveichik. 2017. Enzyme-free nucleic acid dynamical systems. *Science* 358, 6369 (2017).
- [52] Marko Vasić, Cameron Chalk, Austin Luchsinger, Sarfraz Khurshid, and David Soloveichik. 2022. Programming and training rate-independent chemical reaction networks. *Proc. Nat. Acad. Sci.* 119, 24 (2022), e2111552119.
- [53] Günter M. Ziegler. 1995. Lectures on Polytopes. Springer-Verlag, New York.

Received 11 August 2021; revised 10 October 2022; accepted 24 February 2023